AFRL-VA-WP-TR-2002-3053

IMPLEMENTING FAMILIES OF IMPLICIT CHEBYSHEV METHODS WITH EXACT COEFFICIENTS FOR THE NUMERICAL INTEGRATION OF FIRST- AND SECOND-ORDER DIFFERENTIAL EQUATIONS

Jason Wm. Mitchell

Control Theory Optimization Branch (AFRL/VACA)
Control Sciences Division
Air Vehicles Directorate
Air Force Research Laboratory, Air Force Material Command
Wright-Patterson Air Force Base, OH 45433-7542

MAY 2002

FINAL REPORT FOR 01 JANUARY 2002 – 01 MAY 2002

Approved for public release; distribution is unlimited.

AIR VEHICLES DIRECTORATE
AIR FORCE RESEARCH LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542

NOTICE

This technical report has been reviewed and is approved for publication.

Jason W. Mitchell Visiting Scientist Control Theory Optimization Branch	Siva S. Banda Senior Scientist Control Theory Optimization Branch

John A Bowlus

JOHN A. BOWLUS
CHIEF, CONTROL SCIENCES DIVISION
AIR VEHICLES DIRECTORATE

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE ***(DD-MM-YY)***	**2. REPORT TYPE**	**3. DATES COVERED** ***(From - To)***
May 2002	Final	01/01/2002 – 05/01/2002

4. TITLE AND SUBTITLE IMPLEMENTING FAMILIES OF IMPLICIT CHEBYSHEV METHODS WITH EXACT COEFFICIENTS FOR THE NUMERICAL INTEGRATION OF FIRST- AND SECOND-ORDER DIFFERENTIAL EQUATIONS	**5a. CONTRACT NUMBER** IN-HOUSE **5b. GRANT NUMBER** **5c. PROGRAM ELEMENT NUMBER** 61102F
6. AUTHOR(S) Jason Wm. Mitchell	**5d. PROJECT NUMBER** 2304 **5e. TASK NUMBER** N2 **5f. WORK UNIT NUMBER** 01
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Control Theory Optimization Branch (AFRL/VACA) Control Sciences Division Air Vehicles Directorate Air Force Research Laboratory, Air Force Material Command Wright-Patterson Air Force Base, OH 45433-7542	**8. PERFORMING ORGANIZATION REPORT NUMBER** AFRL-VA-WP-TR-2002-3053
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AIR VEHICLES DIRECTORATE AIR FORCE RESEARCH LABORATORY AIR FORCE MATERIEL COMMAND WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542	**10. SPONSORING/MONITORING AGENCY ACRONYM(S)** AFRL/VACA **11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)** AFRL-VA-WP-TR-2002-3053

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT ***(Maximum 200 Words)***

A method is presented for the generation of exact numerical coefficients found in two families of implicit Chebyshev methods for the numerical integration of first- and second-order ordinary differential equations. The exact coefficients are represented as algebraic numbers, those numbers that solve polynomial equations with rational coefficients, and are written in nested radical form when possible. In addition to the integration weights, the intra-nodal locations required by the implicit methods are computed. These exact coefficients provide a convenient way to obtain accurate floating-point weights for an integration method of given degree with respect to a specific floating-point precision.

15. SUBJECT TERMS

Numerical Methods, Numerical Integration, Chebyshev Integration, Implicit Methods

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT:**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** (Monitor)
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	SAR	210	Jason Wm. Mitchell
Unclassified	Unclassified	Unclassified			**19b. TELEPHONE NUMBER** *(Include Area Code)* (937) 255-8686

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18

Contents

1 Background **1**

1.1 Introduction . . . 1

1.1.1 Chebyshev polynomial interpolation . . . 1

1.1.2 Implicit methods . . . 2

1.2 Review of Methods . . . 2

1.2.1 First-order differential equations . . . 2

1.2.2 Second-order differential equations of special form . . . 3

1.2.3 Iteration procedure . . . 4

1.2.4 Method degree recommendations . . . 5

1.3 Goals . . . 6

2 Computing the Exact Coefficients **7**

2.1 Introduction . . . 7

2.2 Mathematical definitions . . . 7

2.2.1 Interval related quantities . . . 8

2.2.2 Double prime summation . . . 8

2.2.3 Finite-sum Chebyshev approximation coefficients . . . 8

2.2.4 Abstracted summation evaluator . . . 8

2.2.5 Generic method generator for specific degree . . . 9

2.2.6 First-order system specifics . . . 9

2.2.7 Second-order special form system specifics . . . 10

2.3 Simplification engine . . . 11

2.4 Utility function definitions . . . 16

2.4.1 Assertion device . . . 16

2.4.2 Constructing arbitrary symbol tags . . . 17

2.4.3 Expression form predicate . . . 17

2.4.4 Special sorting predicate . . . 17

2.4.5 Weighted leaf count . . . 18

2.4.6 Mapping $\cos n\pi/m$ to $\mathbb{A}$ . . . 18

2.4.7 Opening specially named files . . . 19

2.5 LaTeX output functions . . . 19

2.5.1 `longtable` header . . . 20

2.5.2 Intra-nodal stepsizes and `Root` objects . . . 20

2.5.3 Tables of coefficients . . . 21
2.5.4 List of equations . . . 23
2.5.5 LaTeX production public export . . . 24
2.6 Root code chunks . . . 24
2.6.1 Root chunk for family of first-order methods . . . 24
2.6.2 Root chunk for family of special form second-order methods . . . 24
2.6.3 Sample driver script for family of first-order methods . . . 25
2.7 Miscellaneous chunks . . . 25

3 Example for First-Order Systems 31
3.1 Introduction . . . 31
3.2 Basic Requirements . . . 31
3.3 Root chunk: `slice_iterator` . . . 32
3.3.1 Class declaration . . . 32
3.3.2 Member definitions . . . 36
3.4 Root chunk: `default_guesser` . . . 41
3.4.1 Class declaration . . . 42
3.4.2 Member definitions . . . 44
3.5 Includes & header . . . 46
3.6 Root chunk: `pr_constants` . . . 47
3.6.1 Traits template declaration . . . 48
3.6.2 Explicit specialization for `double` . . . 49
3.6.3 Includes & headers . . . 53
3.7 Root chunk: `pr_cheby` . . . 54
3.7.1 Class declaration . . . 54
3.7.2 Member definitions . . . 58
3.7.3 Includes & header . . . 64
3.8 Testing . . . 66
3.8.1 Harmonic oscillator . . . 66
3.8.2 Fourth-order Runge-Kutta method . . . 67
3.8.3 Chebyshev integrator of degree $n = 8$ . . . 71

A Intra-nodal points 75

B Coefficients for first-order methods 80

C Coefficients for second-order methods 138

List of Tables

A.1 Intra-nodal points ξ_j, $n = 2$. 75
A.2 Intra-nodal points ξ_j, $n = 4$. 76
A.3 Intra-nodal points ξ_j, $n = 6$. 76
A.4 Intra-nodal points ξ_j, $n = 8$. 76
A.5 Intra-nodal points ξ_j, $n = 10$. 77
A.6 Intra-nodal points ξ_j, $n = 12$. 77
A.7 Intra-nodal points ξ_j, $n = 14$. 78
A.8 Unique `Root` objects $n = 14$. 78

B.1 Integral approximation I_j for $n = 2$. 80
B.2 Integral approximation I_j for $n = 4$. 81
B.3 Integral approximation I_j for $n = 6$. 82
B.4 Integral approximation I_j for $n = 8$. 83
B.5 Integral approximation I_j for $n = 10$. 90
B.6 Integral approximation I_j for $n = 12$. 102
B.7 Integral approximation I_j for $n = 14$. 114

C.1 Integral approximation I_j for $n = 2$. 138
C.2 Integral approximation I_j for $n = 4$. 139
C.3 Integral approximation I_j for $n = 6$. 140
C.4 Integral approximation I_j for $n = 8$. 141
C.5 Integral approximation I_j for $n = 10$. 148
C.6 Integral approximation I_j for $n = 12$. 160
C.7 Integral approximation I_j for $n = 14$. 172

List of Symbols

(a, b)	The open interval a, b with $b > a$
$[a, b]$	The closed interval a, b with $b > a$
$\|x\|$	The absolute value of x
$\mathbb{A}$	The field of numbers that solve polynomial equations with rational coefficients
$\mathbb{R}$	The field of real numbers
$\mathbb{Z}$	The field of integers
C^n	The space of continuous function with differentiability n
$\sum''$	A summation with the first and last terms halved
$\mathbf{f}(x)$	A vector function of x
$\mathbf{x}$	A vector quantity
$f(x)$	A scalar function of x
h	An interval fixed stepsize
x	A scalar quantity

Acknowledgement

This work was performed while the author held a National Research Council (NRC) Research Associateship award at the Air Force Research Laboratory (AFRL) Air Vehicles directorate in the Control Theory Optimization branch (VACA). The author would like to thank Dr. Siva Banda and Dr. Andrew Sparks for their support and the opportunity to perform this work.

Chapter 1

Background

1.1 Introduction

In this report, we present a method to generate the exact coefficients necessary to implement two families of implicit Chebyshev methods for numerical integration. The first family is valid for systems of first-order ordinary differential equations as presented by Richardson and Panovsky [7]. The second family is applied to systems of second-order ordinary differential equations of a special form as presented by Panovsky and Richardson [5].

Before we begin the brief review of the two families of numerical methods, we address two preliminary topics related to the implementation of these methods.

1.1.1 Chebyshev polynomial interpolation

For the numerical methods discussed here, the interpolation over a particular interval is accomplished using Chebyshev polynomials. These polynomials are a clear choice of basis functions because of their minimax error properties as well as their excellent convergence characteristics. As a result, for a sample interval $[x, x+h]$, the interpolation points or nodes are not evenly spaced along the x-axis. However, they are evenly spaced along the arc-length of a semi-circle centered at the interval midpoint. As an example, Figure 1.1 demonstrates this by dividing the semi-circle into six equal sectors. Each angular position $\theta_j, j \in \mathbb{Z}[0, n]$ is given as

$$\theta_j = \frac{n-j}{n}\pi. \tag{1.1}$$

Considering the unit semi-circle, the locations of the nodal points inside the unit interval, or unit *intra*-nodal points ξ_j, are given by

$$\xi_j = \frac{1}{2}\left(1 + \cos\theta_j\right). \tag{1.2}$$

Conveniently, the above expression also correctly includes the interval end points for $j = 0, n$ so that $\xi_j \in \mathbb{R}[0, 1]$. Returning to the interval $[x, x+h]$, these intra-nodal

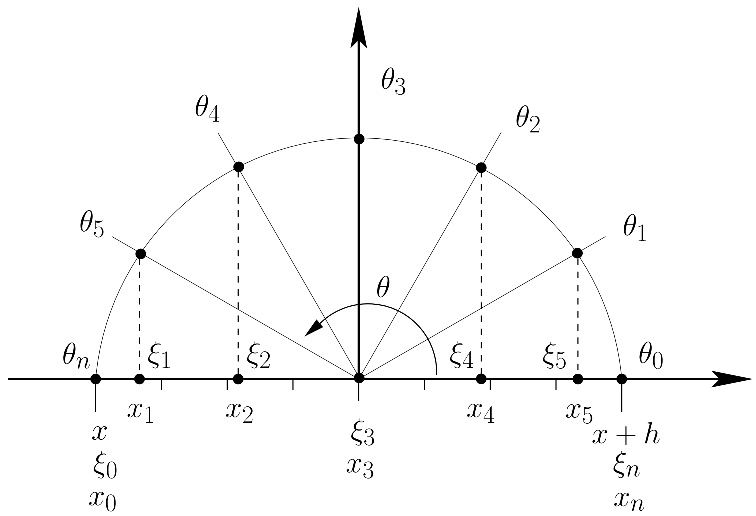

Figure 1.1: Sample interval $[x, x+h]$ for $n = 6$.

points x_j are given by

$$x_j \ = x + h\xi_j. \tag{1.3}$$

For the example given in Figure 1.1, the upper limit of j is $n = 6$. In a terminology consistent with Henrici [3], this upper bound n is called the *degree* of the selected interpolation method.

1.1.2 Implicit methods

The implicit nature of these methods arises as the result of interpolation mechanism used to propagate a given numerical solution $y(x)$ to $y(x+h)$. Obviously, we are interested in $y(x+h)$, i.e. $\xi = 1$. However, to compute $y(x+h)$, we must have information about the values of both the derivative and solution at the at the intra-nodal points. The fact that the derivatives are direct functions of the solution gives rise to the *implicit* nature of these methods.

1.2 Review of Methods

1.2.1 First-order differential equations

For the case of a family of implicit Chebyshev methods for the numerical integration of first-order differential equations as presented by Richardson and Panovsky [7], the form of the initial value problem of dimension m is given by

$$\frac{\mathrm{d}\mathbf{y}}{\mathrm{d}x} = \mathbf{f}(\mathbf{y}(x), x)\,, \quad \mathbf{y}(x)|_{x_0} = \mathbf{y}_0, \quad \mathbf{y}, \mathbf{f} \in \mathbb{R}^m, \tag{1.4}$$

with each $f \in C^1[0, b]$.

The fundamental final result necessary to develop the integration procedure for a specific m is given by

$$y(x + h\xi_j) = y(x) + \frac{1}{2}h \sum_{k=0}^{n}{}'' a_k R_{jk}, \tag{1.5}$$

with the summand terms defined as

$$a_k = \frac{2}{n} \sum_{j=0}^{n}{}'' f(x_j) \cos k\theta_j, \tag{1.6}$$

and

$$R_{jk} = \begin{cases} \dfrac{\cos(k+1)\theta_j + (-1)^k}{2(k+1)} - \dfrac{\cos(k-1)\theta_j + (-1)^k}{2(k-1)}, & k \geq 2, \\ \dfrac{1}{4}\left(\cos 2\theta_j - 1\right), & k = 1, \\ \cos\theta_j + 1, & k = 0. \end{cases} \tag{1.7}$$

The local truncation error for a given degree n induced by the discrete approximation found in Eq. (1.5) is written $e_n(z)$, where

$$z = x + h\xi, \tag{1.8}$$

with $\xi \in \mathbb{R}[0,1]$. For $f(\xi)$ evaluated at some point $\beta \in \mathbb{R}[0,1]$, the error estimate is given by

$$e_n(z) \approx \begin{cases} \dfrac{nh^{n+3} D^{n+2} f\left(z(\beta)\right)}{2^{2n}(n^2-1)(n^2-9)(n+1)!}, & n \text{ even}, \\ \dfrac{h^{n+2} D^{n+1} f\left(z(\beta)\right)}{2^{2n-1} n(n^2-4)(n+1)!}, & n \text{ odd}, \end{cases} \tag{1.9}$$

where $D^k f\left(z(\beta)\right)$ represents the k^{th} derivative of f evaluated at $z(\beta)$.

1.2.2 Second-order differential equations of special form

For the case of a family of implicit Chebyshev methods for the numerical integration of second-order differential equations, as presented by Panovsky and Richardson [5], the special form of the initial value problem of dimension m is given by

$$\frac{\mathrm{d}^2\mathbf{y}}{\mathrm{d}x^2} = \mathbf{f}(\mathbf{y}(x), x), \quad \mathbf{y}(x)|_{x_0} = \mathbf{y}_0, \quad \mathbf{y}'(x)|_{x_0} = \mathbf{y}'_0, \quad \mathbf{y}, \mathbf{y}', \mathbf{f} \in \mathbb{R}^m, \tag{1.10}$$

with each $f \in C^2[0,b]$. Examining Eq. (1.10), we quickly notice that no first derivatives appear in the arguments of $\mathbf{f}$, thus constituting the special form. Panovsky and Richardson [5] provide a discussion for the range of applications of this problem form.

The fundamental final result necessary to develop the integration procedure for a specific m is given by

$$y(x+h\xi_j) \;= 2y(x) - y(x-h\xi_j) + \frac{1}{4}h^2 \sum_{k=0}^{n}{}'' (a_k^+ + a_k^-) S_{jk}, \tag{1.11}$$

with the summand terms

$$a_k^{\pm} \;= \frac{2}{n} \sum_{j=0}^{n}{}'' f(x_j^{\pm}) \cos k\theta_j, \tag{1.12}$$

where $x_j^{\pm} = x \pm h\xi_j$, and

$$S_{jk} \;= \begin{cases} \dfrac{\cos{(k+2)}\,\theta_j}{4(k+1)(k+2)} + \dfrac{\cos{(k-2)}\,\theta_j}{4(k-1)(k-2)} - \dfrac{\cos k\theta_j}{2(k+1)(k-1)} + & \\ \quad + \dfrac{(-1)^{k+1}\cos\theta_j}{(k+1)(k-1)} + \dfrac{(-1)^{k+1}}{(k+2)(k-2)}, & k \geq 3^{\dagger}, \\ \dfrac{1}{48}\left(-9 - 16\cos\theta_j - 8\cos 2\theta_j + \cos 4\theta_j\right), & k = 2, \\ \dfrac{1}{24}\left(8 - 9\cos\theta_j + \cos 3\theta_j\right), & k = 1, \\ \dfrac{1}{4}\left(3 - 4\cos\theta_j + \cos 2\theta_j\right), & k = 0. \end{cases} \tag{1.13}$$

As before, the local truncation error for a given degree n incurred by the discrete approximation found in Eq. (1.11) is written $e_n(z)$ for z given by Eq. (1.8) with $\xi \in \mathbb{R}[-1,1]$. Then, for $f(\xi)$ evaluated at some point $\beta \in \mathbb{R}[-1,1]$, the error estimate is written as

$$e_n(z) \;\approx \frac{h^{n+3}}{2^{2n-1} n^3 (n+1)!} \left| D^{n+1} f\left(z(\beta)\right) \right|. \tag{1.14}$$

The above error estimate presumes that

$$\left| D^{n+1} f^{+}(\sigma) \right| \;\approx \left| D^{n+1} f^{-}(\eta) \right| \;\approx\; \left| D^{n+1} f(\beta) \right|, \tag{1.15}$$

for $\sigma, \eta \in \mathbb{R}[-1,1]$.

1.2.3 Iteration procedure

For the family of methods discussed in § 1.2.2, Eq. (1.11) indicates that intra-nodal information from the previous step is required, viz. a_k^-. That information can be stored from the previous iteration and easily made available. Additionally, it indicates that this family of methods is not self-starting. To provide the necessary starting

†For $k \geq 3$, S_{jk} appearing here typographically corrects R_{jk} given by Eq. (21) in Panovsky and Richardson [5, p. 39]; cf. Carpenter [2, p. 56].

Algorithm 1.1 Example continuous substitution iteration

compute intra-nodal step sizes: $h_j \leftarrow x_j - x_{j-1}$
while $x < x_f$ **do**
  for all $j \in \mathbb{Z}[1, n]$ **do**
    generate initial guesses for all y_j: $\mathbf{y}^{(0)}$
    repeat
      compute $a_k^{(\ell)}$
      compute $y_j^{(\ell)}$
    until $\|\mathbf{y}^{(\ell)} - \mathbf{y}^{(\ell-1)}\| < \epsilon$
  end for
  $\mathbf{y}^{(\ell-1)} \leftarrow \mathbf{y}^{(\ell)}$
  increment x
end while

information any number of high-accuracy methods may be used. The fact that high-accuracy, fully explicit or predictor-corrector methods are computationally expensive is of little concern because it is only required to bootstrap the process. Fortunately, the family of methods for first-order systems (§ 1.2.1) is self-starting, as indicated by Eq. (1.5).

Despite their differences, both families of methods still share similar iteration schemes because derivative and solution values at the forward intra-nodal points must be known. To enhance the process, it is convenient to generate an initial guess with a simple, low-order, fully explicit method. Experimentation has shown that a fourth-order Runge-Kutta scheme is sufficient for this task. Once the initial guesses have been obtained for each y_j, a method termed *iteration by continuous substitution*[1] is used to until the difference between two successive y_j is sufficiently small. An example algorithm for this is given in Algorithm 1.1. This sample algorithm is by no means optimized for speed of execution, but rather illustrative of the procedure necessary to implement the families of methods.

1.2.4 Method degree recommendations

For second-order systems of special form, the local truncation error analysis performed in Panovsky and Richardson [5] does not explicitly state any recommendations for choice of method degree with respect to computational efficiency. However, the analysis presented in Richardson and Panovsky [7] for first-order systems clearly favors even method degrees since for a given degree n even, the truncation errors for n and $n + 1$ are very similar. This is clearly seen from Eq. (1.9).

[1] See Panovsky and Richardson [5] for additional detail.

1.3 Goals

Richardson et al. [8] provided several significant performance improvements in the implementation of the family of first-order methods. Further significant performance improvement for both families could be obtained by expanding the coefficients a_k and $a_k^{\pm}$ directly into Eqs. (1.5) and (1.11), respectively. This would eliminate their explicit computation in the inner-most iteration of Algorithm 1.1 and considerably speed the process. However, as Panovsky and Richardson [5, p. 41] noted, "The explicit replacement of [these coefficients] ... would require considerable algebraic manipulations even for modest values of n." Indeed, considerable effort would be required to perform this algebra by hand for a single interesting value of n. Fortunately, symbolic algebra systems allow us to perform algebraic operations that would ordinarily be most unproductive if performed by hand. To accomplish this task, we develop a package using the MATHEMATICA® symbolic algebra system.

In addition to automatically performing large quantities of tedious algebra, there are additional benefits to such an approach. One obvious benefit is that the required coefficients can be computed exactly as algebraic numbers[2]. Knowing the coefficients exactly will ensure that they can be computed externally to any accuracy desired for a given precision.

In terms of the explicit forms for first- and second-order methods, we want to write Eqs. (1.5) and (1.11) as

$$y(x + h\xi_j) \ = y(x) + h \sum_{k=0}^{n} w_{jk}\, f(x_k), \tag{1.16}$$

and

$$y(x + h\xi_j) \ = 2y(x) - y(x - h\xi_j) + h^2 \sum_{k=0}^{n} w_{jk} \left(f(x_k^{+}) + f(x_k^{-}) \right), \tag{1.17}$$

respectively. In each of the about equations, we seek to compute the w_{jk} exactly.

[2]See Wolfram [10, p. 1070].

Chapter 2

Computing the Exact Coefficients

2.1 Introduction

As mentioned in the previous chapter, MATHEMATICA® was selected to perform as the symbolic algebra engine. It is certainly not the only symbolic algebra system that could have been selected. Many other systems exist that could perform this task.

In the following sections, we develop and describe the MATHEMATICA® code necessary to compute the exact coefficients for both families of implicit methods previously discussed. To better document the program development, `noweb`[1] by Ramsey [6] was used. The coefficients themselves appear in appendices A–C. For notational convenience, all derivative evaluations are written $f(x)$ rather than the fully explicit $f(y(x), x)$. Thus $y(x)$ is always an implicit argument when writing $f(x)$.

We begin by considering the mathematics from Chapter 1.

2.2 Mathematical definitions

The common mathematics from Chapter 1 are

7 ⟨*mathematical definitions: common* 7⟩≡ (24)
  ⟨*define interval related quantities* 8a⟩
  ⟨*define double prime summation* 8b⟩
  ⟨*define finite-sum Chebyshev approximation coefficients* 8c⟩
  ⟨*define abstracted summation evaluator* 8d⟩
  ⟨*define generic method generator for given degree* 9a⟩

with family specific mathematics defined in code chunks that follow.

[1]The `noweb` system is a considerably simplified implementation of the *literate programming* concept presented by Knuth [4].

2.2.1 Interval related quantities

From § 1.1.1, the location of the points of interest on the unit interval from Eqs. (1.1) and (1.3) become

8a ⟨*define interval related quantities* 8a⟩≡ (7)

```
theta[ j_Integer, n_Integer ] := (n-j) Pi / n;

xi[j_Integer, n_Integer] := ( 1 + Cos[ theta[j,n] ] ) / 2;
```

2.2.2 Double prime summation

8b ⟨*define double prime summation* 8b⟩≡ (7)

```
sumpp[ elist_List ] :=
  Module[ {},
    Return[ Plus @@  MapAt[#/2&, elist, { {1}, {-1} }] ];
  ];
```

2.2.3 Finite-sum Chebyshev approximation coefficients

Eqs. (1.6) and (1.12) provide the finite-sum approximation Chebyshev coefficients for each family. Since the a_k from § 1.2.1 are the same as the a_k^+ from § 1.2.2, the definition of $a_k^\pm$ is used to provide a_k.

8c ⟨*define finite-sum Chebyshev approximation coefficients* 8c⟩≡ (7)

```
ak[ k_Integer /; k >= 0, s_Integer /; s === +1|| s === -1,
    n_Integer /; n > 0 ] :=
  Module[ {elist, j},
    elist = Table[ f[x + Sign[s] h xi[j,n] ] Cos[k theta[j,n] ],
                   {j,0,n} ];
    Return[ (2/n) sumpp[elist] ];
  ];
```

The second argument to `ak[...]`, i.e. `s`, represents the sign required to compute each x_j. As the condition indicates, `s` must be given such that $\texttt{s} = \pm 1$.

2.2.4 Abstracted summation evaluator

8d ⟨*define abstracted summation evaluator* 8d⟩≡ (7)

```
summation[ j_Integer /; j > 0, n_Integer /; n > 0 ] :=
  Module[{},
    pickupGenOptions[ Equations, opts ];
    Return[ sumpp[ sumtable[j, n] ] ];
```

```
  ];
```

For the summations appearing in Eqs. (1.5) and (1.11), each of the terms are generated and placed in a `Table` with the first and last entries of the list halved (`sumpp`), as required, and then added together. The `sumtable` and leading multipliers of the summations are handled in family specific code chunks that follow.

2.2.5 Generic method generator for specific degree

For both families of methods, the method for a particular degree is first generated in raw form, and will require considerable simplification.

9a ⟨*define generic method generator for given degree* 9a⟩≡ (7)

```
GenEqsRaw[ n_Integer /; n > 0 && EvenQ[n], opts___Rule ] :=
  Module[ {j},
    pickupGenOptions[ Equations, opts ];

    Return[ Table[ { y[ x + h xi[j,n] ],
                     leadingterms[j,n],
                     summation[j,n] },
                   {j,n} ] ];
  ];
```

In the table returned from `GenEqsRaw`, the first list element of each table entry is the intra-nodal point of interest. The second and third elements of the list contain the corresponding leading terms and summation information, respectively, from Eqs (1.5) and (1.11). This information is family specific.

2.2.6 First-order system specifics

Here, we address the family specific mathematics from § 1.2.1.

9b ⟨*mathematical definitions:* $\mathcal{O}(1)$ 9b⟩≡ (24b) 10a ▷

```
R[ j_Integer /; j > 0, k_Integer /; k >= 0,
   n_Integer /; n > 0 ] :=
  Module[ {tj},
    tj=theta[j,n];
    Which[
      k == 0,
        (Cos[tj] + 1),
      k == 1,
        (Cos[2 tj] - 1) / 4,
      True, (* k >= 2 *)
        ( (Cos[ (k+1)tj ] + (-1)^k) / (k+1) -
          (Cos[ (k-1)tj ] + (-1)^k) / (k-1)  ) / 2
```

```
    ]
  ];
```

The above chunk provides R_{jk} from Eq. (1.7).

10a ⟨*mathematical definitions:* $\mathcal{O}(1)$ 9b⟩+≡ (24b) ◁9b 10b▷

```
sumcoefficient = {h, 2};
```

`sumcoefficient` defines the leading multiplier of the summation of Eq. (1.5).

10b ⟨*mathematical definitions:* $\mathcal{O}(1)$ 9b⟩+≡ (24b) ◁10a 10c▷

```
leadingterms[ j_Integer, n_Integer ] := (y[x]);
```

`leadingterms` represents the non-summation terms in the right hand side of Eq. (1.5).

10c ⟨*mathematical definitions:* $\mathcal{O}(1)$ 9b⟩+≡ (24b) ◁10b

```
sumtable[ j_Integer /; j > 0, n_Integer /; n > 0 ] :=
  Table[ ak[k,+1,n] R[j,k,n], {k,0,n} ];
```

Lastly, `sumtable` provides the summand information for Eq. (1.5).

2.2.7 Second-order special form system specifics

Here, we address the family specific mathematics from § 1.2.2.

10d ⟨*mathematical definitions:* $\mathcal{O}(2)s$ 10d⟩≡ (24c) 11a▷

```
S[ j_Integer /; j > 0, k_Integer /; k >= 0,
   n_Integer /; n > 0 ] :=
  Module[ {tj},
    tj=theta[j,n];
    Which[
      k == 0,
        1/4( 3 + 4 Cos[tj] + Cos[2 tj] ),
      k == 1,
        1/24( -8 - 9 Cos[tj] + Cos[3 tj]),
      k == 2,
        1/48( -9 - 16 Cos[tj] - 8 Cos[2 tj] + Cos[4 tj]),
      True, (* k > 2 *)
        Cos[ (k+2)tj ] / ( 4(k+1)(k+2) ) +
        Cos[ (k-2)tj ] / ( 4(k-1)(k-2) ) -
        (Cos[ k tj ] + 2 Cos[ k Pi ] Cos[ tj ]) / ( 2(k-1)(k+1) ) -
        Cos[ k Pi] / ( (k-2)(k+2) )
    ]
  ];
```

The above chunk defines S_{jk} from Eq. (1.13).

11a $\langle$*mathematical definitions: $\mathcal{O}(2)s$* 10d$\rangle{+}{\equiv}$ (24c) $\triangleleft$10d 11b$\triangleright$

```
sumcoefficient = {h^2, 4};
```

`sumcoefficient` provides the leading multiplier of the summation of Eq. (1.11).

11b $\langle$*mathematical definitions: $\mathcal{O}(2)s$* 10d$\rangle{+}{\equiv}$ (24c) $\triangleleft$11a 11c$\triangleright$

```
leadingterms[ j_Integer, n_Integer /; n > 0 ] :=
  (2 y[x] - y[x - 1/2 (1+Cos[(n-j) Pi/n]) h ]);
```

`leadingterms` represents the non-summation terms in the right hand side of Eq. (1.11).

11c $\langle$*mathematical definitions: $\mathcal{O}(2)s$* 10d$\rangle{+}{\equiv}$ (24c) $\triangleleft$11b

```
sumtable[ j_Integer /; j > 0, n_Integer /; n > 0 ] :=
  Table[ (ak[k,+1,n]+ak[k,-1,n]) S[j,k,n], {k,0,n} ];
```

Lastly, `sumtable` computes the summand information for Eq. (1.11).

2.3 Simplification engine

`Equations` is the publicly exported function that users invoke to obtain the method of degree n for a given family. This function does not generate the raw equations directly, `GenEqsRaw` performs that task. However, it does perform the specific simplifications necessary to obtain the most compact form of a method.

11d $\langle$*simplification engine: common:head* 11d$\rangle\equiv$ (24)

```
Equations[ n_Integer /; n > 0 && EvenQ[n], opts___Rule ] :=
```

$\langle$*state module decls and process user opts* 11e$\rangle$
$\langle$*generate raw degree n equations* 12a$\rangle$
$\langle$*apply $g : \cos n\pi/m \mapsto \mathbb{A}$ or apply* `N` 12b$\rangle$
$\langle$*replace numeric stepsizes in function args with symbols* 12c$\rangle$
$\langle$*fully simplify expressions to get single weight for each* `f` 14c$\rangle$

The additional argument predicate `EvenQ` attached to `n` ensures that only even degree methods will be generated as recommended in § 1.2.4. For convenience, methods for second-order systems of special form are also so constrained. However, simply removing the predicate here and in `GenEqsRaw` will permit generation of methods of odd degree.

11e $\langle$*state module decls and process user opts* 11e$\rangle\equiv$ (11d)

```
  Module[ {j, isExpandFailed=False, creplist={}, flist, hlist,
           hrlist, frlist, tmp, den, coef, result},
    pickupGenOptions[ Equations, opts ];
```

12a ⟨*generate raw degree n equations* 12a⟩≡ (11d)
```
Print[ "n = ", n, ":"];
Print[ " Generating equations."];
eqs = GenEqsRaw[ n, opts ];
```

12b ⟨*apply* $g : \cos n\pi/m \mapsto \mathbb{A}$ *or apply* `N` 12b⟩≡ (11d)
```
If[ exactp == True,
  Print[ "  Expanding Cos[n Pi / m] terms."];
  creplist = CosRepList[ eqs ];
  If[ Not[ IsDesiredForm[ creplist ] ] ,
    Print["  ...expansion failed."];
    $MaxExtraPrecision += 50;
    isExpandFailed = True;
  ];,
(*else*)
  Print[ "  Converting to numerical coefficients, prec = ",prec];
  eqs = Simplify[ N[ eqs, prec ] ];
];
```

If the replacement fails, the boolean `isExpandFailed` is set to `True`. Failure is measured against `IsDesiredForm` which requires that the expressions be free of `Root`, `Complex`, and `Power[-1,_]` objects.

12c ⟨*replace numeric stepsizes in function args with symbols* 12c⟩≡ (11d)
⟨*extract unique function calls* 12d⟩
⟨*build replacement rule for forward unit intra-nodal points* 12e⟩
⟨*handle replacement for back intra-nodal points* 13a⟩
⟨*perform stepsize replacement* 13b⟩
⟨*check if* $\mathbb{A}$ *numbers are irreducible* `Root` *objects* 13c⟩

12d ⟨*extract unique function calls* 12d⟩≡ (12c)
```
Print[ "  Simplifying intra-nodal stepsizes."];
flist = Cases[ eqs, _[_], Infinity] // Union;
```

12e ⟨*build replacement rule for forward unit intra-nodal points* 12e⟩≡ (12c)
```
hlist = Table[ xi[j,n], {j,1,n-1}] h;
hrlist = Table[ hlist[[j]] -> MakeSym[h,j], {j,Length[hlist]} ]
         // Sort[ #, Greater[ LeafCount[#1], LeafCount[#2] ]& ]&;
frlist = flist // ExpandAll;
```

The `Sort` applied to `hrlist` uses `LeafCount` to provide a sort based on expression complexity. The purpose is to move the more complex expressions to the head of the list, so that less complex expressions are not incorrectly substituted into portions of the more complex expressions. For example, for the two rules `x3->a, x4->a+b}`, the

rule for `x4` must be executed first to avoid the faulty execution of rule `x3` resulting in incorrect expressions, viz. `x3+b` rather than simply `x4`.

13a ⟨*handle replacement for back intra-nodal points* 13a⟩≡ (12c)
```
For[ j = 1, j <= Length[hrlist], j++,
  frlist = frlist //. ExpandAll[{ hrlist[[j]],
                          Rule @@ (-#& /@ List @@ hrlist[[j]]) }]
];
```

The purpose of the `-#&` rule is to also replace the any back intra-nodal points with negative version of the intra-nodal stepsize, i.e. $-h_j$. This does not affect the family of first-order methods, since back points will not exist in the function arguments, so it is safe to apply in both cases.

13b ⟨*perform stepsize replacement* 13b⟩≡ (12c)
```
hrlist[[All,1]] = hrlist[[All,1]] // Collect[#, h, Simplify] &;
frlist = Thread[ Rule[flist, frlist] ];
eqs = (eqs //. frlist);
```

If the resulting expression does not meet the desired form, e.g. contains `Root` objects because the resulting polynomial equations are not reducible to nested square roots, a different method of labeling the `Root` objects is invoked.

13c ⟨*check if* $\mathbb{A}$ *numbers are irreducible* `Root` *objects* 13c⟩≡ (12c)
```
If[ isExpandFailed,
  ⟨grab all Cos occurrences 13d⟩
  ⟨compute maximally reduced Root objects 13e⟩
  ⟨calculate unique numerical values with extra precision 13f⟩
  ⟨construct replacement rule for (numerically) unique Root objects 14a⟩
];
⟨apply Root replacement rule 14b⟩
```

13d ⟨*grab all* `Cos` *occurrences* 13d⟩≡ (13c)
```
  Print[ "  ...labeling/replacing Root[] objects."];
  coslist = Cases[ eqs, Cos[_], Infinity] // Union;
```

13e ⟨*compute maximally reduced* `Root` *objects* 13e⟩≡ (13c)
```
  rootlist = coslist // TrigToExp // RootReduce // FullSimplify;
```

13f ⟨*calculate unique numerical values with extra precision* 13f⟩≡ (13c)
```
  nlist = Abs[N[rootlist, $MaxExtraPrecision]]
          // Union[#,
             SameTest->(Abs[#1-#2]<10^(-$MaxExtraPrecision+1)&)]&;
```

The `SameTest` used for the `Union` attempts to provide sufficient precision to resolve roots that may be close together. This is most likely not necessary since the

Chebyshev intra-nodal points should be sufficiently different to avoid this problem for most degrees n of interest. However, for very large n, intra-nodal clustering near the interval end points may cause difficulty depending on the required floating-point precision.

An interesting side effect of this `SameTest` is that it accounts for a slight difference in the way `N` is applied between MATHEMATICA® versions 4.0 and 4.1.

14a ⟨*construct replacement rule for (numerically) unique* `Root` *objects* 14a⟩≡ (13c)

```
clist = Table[ MakeSym[c,j], {j,Length[nlist]} ];
nclist = Thread[Rule[nlist,clist]];
nclist = {Table[ Rule @@ (-#& /@ List@@nclist[[k]]),
                 {k,Length[nclist]}],
          nclist} //Flatten;
rclist = Thread[Rule[rootlist,
                     (N[rootlist,$MaxExtraPrecision]/.nclist)]];
creplist = Thread[ Rule[ coslist, (rootlist//.rclist) ] ];
```

The meaning of the above code chunk may be difficult to perceive, initially. Its goal is to determine which `Root` objects are unique (numerically), symbolically label them (c1,c2,...), replace repeated `Root` objects with the appropriate label, then replace all `RootReduce`'d `Cos` expressions with the appropriate symbol.

The resulting replacement rules are only applied if `creplist` is not empty.

14b ⟨*apply* `Root` *replacement rule* 14b⟩≡ (13c)

```
If[ creplist != {},
  eqs = eqs //. creplist;
];
```

14c ⟨*fully simplify expressions to get single weight for each* `f` 14c⟩≡ (11d)

```
Print[ "  Factoring common terms from integral approximation."];
WriteString[ $Output, "    j = "];
For[ j = 1, j <= n, j++,
  WriteString[ $Output, j, ", "];
  t1 = (eqs[[j,3]] / sumcoefficient[[2]] ) // ExpandAll
         // Collect[ #, f[_], Simplify ]&
         // Cancel // Together;
  den = t1 // Denominator;
  coef = sumcoefficient[[1]] / den;
  eqs[[j,3]] = coef;
  t1 = t1 // Numerator
          // Collect[#,f[_],
                  FullSimplify[#,ExcludedForms->{Sqrt[_]}]&]&;

  t2 = Cases[ t1, f[_], Infinity ] // Union;
  Assert[ t2[[1]] == f[x], "PRC::expr missing f[b]" ];
  t3 = Coefficient[ t1, t2 ];
```

```
t4 = Thread[ List[t3,t2] ];
```

For the above chunk, the denominator of the summation multiplier for each equation is incorporated into the coefficients and a common denominator factored. The combined summation multiplier is then incorporated into the output list. In finding a common denominator, care is taken not to disturb any `Sqrt` objects appearing.

The family specific code for first-order methods is quite brief. At this stage the simplification is complete. The only remaining task is to rearrange the list slightly so that the coefficients appear from left to right in increasing order of intra-nodal location.

15a ⟨*simplification engine:* $\mathcal{O}(1)$ 15a⟩≡ (24b)

```
t1 = Prepend[ RotateLeft[ Rest[t4], 1], First[t4] ];
```

The family specific code for second-order special form methods is not as brief as for the first-order methods. These expressions contain both forward and back intra-nodal location information. Given the symmetry in the problem, the summand of Eq. (1.11) can be written $w_j\,(f(x+h_j)+f(x-h_j))$. This is accomplished in the `For` block below.

Once the desired summand form is achieved, the expressions still require rearrangement for viewing convenience. This is handled in a fashion similar to the first-order methods. However, there is one notable difference. For the case of second-order special form methods, it is trivial to show that the n^{th} equation of degree n is always a corrector, thus lacking an $f(x+h)+f(x-h)$ term. As a consistency check and for viewing convenience, this is explicitly stated in the output equations.

15b ⟨*simplification engine:* $\mathcal{O}(2)s$ 15b⟩≡ (24c)

```
t5 = Rest[ t4 ] // Partition[#,2]&;
For[ k = 1; m = Length[t5]; t1 = {}, k <= m, k++,
  tmp = t5[[k]];
  If[ SameQ[ tmp[All,1] ],
    AppendTo[ t1, {tmp[[1,1]], Plus@@tmp[[All,2]]} ]; ,
  (*else*)
    Throw["PRC::unexpected intra-nodal asymmetry."];
  ];
];
If[ FreeQ[ t1, f[x+h] ],
  PrependTo[ t1, First[t4] ];
  AppendTo[ t1, {0, f[x-h]+f[x+h]} ];,
(*else*)
  t1 = Prepend[ RotateLeft[t1,1], First[t4] ];
];
```

15c ⟨*simplification engine: common:tail* 15c⟩≡ (24)

```
AppendTo[ eqs[[j]], t1 ];
```

```
    If[ IsDesiredForm[ eqs[[j]] ] === False,
      WriteString[ $Output, "(",j,"x)" ];
    ];
  ];

  hrlist = (Reverse /@ hrlist) //. creplist // Sort[#,noLettersQ]&;
  hrlist[[All,2]] = FullSimplify[ Coefficient[hrlist[[All,2]], h] ] h;
  result = { hrlist, eqs};

  If[ isExpandFailed,
    crlist = Thread[ Rule[ clist, clist /. (Reverse /@ rclist) ] ];
    AppendTo[ result, crlist ];
  ];

  Print["\n Done."];

  Return[ result ];
];
```

2.4 Utility function definitions

16a ⟨*utility functions* 16a⟩≡ (24)

⟨*convenient assertion device* 16b⟩
⟨*construct compound symbols from* `String+Number` *combinations* 17a⟩
⟨*predicate to validate desired expression form* 17b⟩
⟨*predicate to sort* `String` *by numeric values appearing* 17c⟩
⟨*complexity function that provides a weighted leaf count* 18a⟩
⟨*function that constructs and applies* $g : \cos n\pi/m \mapsto \mathbb{A}$ 18b⟩
⟨*function to handle opening output files of form* `n[0-9]+-[ab].tex` 19a⟩

2.4.1 Assertion device

16b ⟨*convenient assertion device* 16b⟩≡ (16a)

```
Assert[ expr_, excpt_:Null ] :=
  Block[{},
    Assert::indet  = "Indeterminate boolean expression `1` encountered.";
    Assert::failed = "Assertion `1` failed.";

    If[ Evaluate[expr],
      Null,                          (* True, do nothing *)
```

```
      Message[Assert::failed, expr]; (* false => fail *)
      Throw[excpt],
      Message[Assert::failed, expr]; (* other => fail differently *)
      Message[Assert::indet, expr];
      Throw[excpt];
    ];
  ];
SetAttributes[Assert, HoldFirst];
```

2.4.2 Constructing arbitrary symbol tags

17a ⟨*construct compound symbols from* `String+Number` *combinations* 17a⟩≡ (16a)

```
MakeSym[q_, p_Integer] :=
  Block[ {res},
    res = MakeExpression[ ToString[q] <> ToString[p], StandardForm];
    Return[ res // ReleaseHold ];
  ];
```

2.4.3 Expression form predicate

The predicate `IsDesiredForm` determines if an input expression is free from complex numbers, `Root` objects, and terms like $(-1)^p$. A result of `False` indicates that the expression is *not* of the desired form.

17b ⟨*predicate to validate desired expression form* 17b⟩≡ (16a)

```
IsDesiredForm[ expr_ ] := ( FreeQ[expr, Power[-1,_], Infinity] &&
                            FreeQ[expr, Complex[_,_], Infinity ] &&
                            FreeQ[expr, Root[__,_], Infinity ] );
```

2.4.4 Special sorting predicate

When sorting certain expressions, it is preferable to avoid lexical sorting and instead ask that the predicate use numerical ordering. The predicate `noLettersQ` does just that. While the code may look rather messy, the fundamental idea is straight forward. When comparing two expressions, the predicate first ensures both are strings, then converts both arguments to character lists and deletes any element matching `LetterQ`. From there, the remaining digits are rejoined as strings and manually converted to expressions that are subject to `OrderQ`. Of course, `noLettersQ` is expecting to sort only on the right hand sides of rules of the form `x1->1` rather than single entries.

17c ⟨*predicate to sort* `String` *by numeric values appearing* 17c⟩≡ (16a)

```
noLettersQ = OrderedQ[ToExpression[((StringJoin@@#&)/@#&)[
               ((DeleteCases[#,_?LetterQ]&)/@#&)[
                (Characters/@#&)[ToString/@((#[[1]]&)/@{##})]]]]]&;
```

2.4.5 Weighted leaf count

The MATHEMATICA® `FullSimplify` function allows users to specify a complexity function to rank the complexity of various expression forms. By default, forms are ranked according to their `LeafCount`. To avoid the expressions specified by `IsDesiredForm`, their `LeafCount` complexity is artificially increased by a factor of 10^{1000}. This approach is a brute-force application of a penalty method to direct the simplification process. It is neither elegant nor efficient, but accomplishes the task in a reasonable time. To keep efficiency issues to a minimum, it is only used in the simplification of expressions of the form `Cos[n Pi/m]`.

18a ⟨*complexity function that provides a weighted leaf count* 18a⟩≡ (16a)

```
WLC[iexpr_] :=
  Module[{fudge = 10^1000, lc = LeafCount[iexpr]},
    If[ Not[ IsDesiredForm[ iexpr ] ],
      lc *= fudge ] ;
    Return[ lc ];
  ];
```

2.4.6 Mapping $\cos n\pi/m$ to $\mathbb{A}$

`CosRepList` attempts to transform expressions of the form `Cos[n Pi/m]` to algebraic numbers, specifically nested square roots, within a given expression list.

18b ⟨*function that constructs and applies* $g : \cos n\pi/m \mapsto \mathbb{A}$ 18b⟩≡ (16a)

```
CosRepList[ eqs_ ] :=
  Module[ {cosines, rlist={}},
    If[ MemberQ[ eqs, Cos[_], Infinity ],
      cosines = Cases[ eqs, Cos[_], Infinity ] // Union;
      rlist = (# -> FullSimplify[TrigToExp[#],
                             ComplexityFunction->WLC]) & /@ cosines;

    ];
    Return[ rlist ];
  ];
```

2.4.7 Opening specially named files

19a ⟨*function to handle opening output files of form* `n[0-9]+-[ab].tex` 19a⟩≡ (16a)

```
myOpenWrite[ str_String, opts___Rule ] :=
  Module[ {filename},
    pickupTeXOptions[ EquationsLaTeXed, opts ];

    If[ StringQ[ofn],
      filename = ofn <> str <> ".tex";
      os = OpenWrite[filename];
      If[ os == $Failed,
        Throw[{myOpenWrite,OpenWrite,filename,$Failed}]];
      Return[ os ],
    (*else*)
      Throw[{myOpenWrite,StringQ[],$Failed}];
    ];
  ];
```

2.5 LaTeX output functions

Producing LaTeX output of the methods generated here requires considerably more processing than the standard `TeXForm` command. The following code chunks address this need.

Essentially two options are available for the LaTeX output: tabular or standard equation form. For the tabular form, tables of coefficients with appropriate table headings, captions and labels are produced. To account for the possibility of having tables that may span multiple pages, the `longtable` package is used for every table. For the case of standard equation form, constants like `Root` objects and intra-nodal stepsizes are placed in standard `amsmath` `align` environments. In both output forms, the `breqn` package[2] is required to automatically and semi-intelligently break multi-line expressions. To use the package, insert

```
\usepackage[cmbase]{flexisym}
\usepackage{breqn}
\setkeys{breqn}{compact}
```

into the preamble of your LaTeX wrapper. Additionally, for tabular output, the command `\mx` must be defined with the form

```
\providecommand{\mx}[1]{\begin{dmath*}[style={\small}]#1\end{dmath*}}
```

for the output to be successfully processed.

[2]This package is available via FTP from `ftp.ams.org/pub/tex/`.

19b ⟨*LaTeX output functions* 19b⟩≡ (24)
⟨*function to generate* `longtable` *header* 20a⟩
⟨*function to LaTeX simple lists of constants* 20b⟩
⟨*function to LaTeX material as coefficient tables* 21⟩
⟨*function to LaTeX material as standard equations* 23⟩
⟨*publicly exported LaTeX output function* 24a⟩

2.5.1 `longtable` header

The table header produced is for use with the `longtable` package.

20a ⟨*function to generate* `longtable` *header* 20a⟩≡ (19b)

```
TableHead[ caption_String, shortcaption_String, head_String ] :=
  StringJoin[
    caption,
    "  \\hline\\hline\n",
    head,
    "  \\hline\n",
    "  \\endfirsthead\n",
    shortcaption, " (\emph{continued})}\\\\\n",
    "  \\hline\\hline\n",
    head,
    "  \\hline\n",
    "  \\endhead\n",
    "  \\hline\n",
    "  \\multicolumn{2}{r}{(\emph{continues})}\\\\\n",
    "  \\endfoot\n",
    "  \\hline\\hline\n",
    "  \\endlastfoot\n"
  ];
```

2.5.2 Intra-nodal stepsizes and `Root` objects

20b ⟨*function to LaTeX simple lists of constants* 20b⟩≡ (19b)

```
texconsts[ eqs_, tabhead_, os_, opts___Rule ] :=
  Module[ {tmp, lhs, rhs, j, n, environ="align",
           endinput="", newline="\\\\"},
    pickupTeXOptions[ EquationsLaTeXed, opts ];

    tmp = (List@@#)& /@ eqs;
    n = Length[ tmp ] + 1;

    If[ isTabular,
```

```
    environ = "longtable";
    cols = "{p{\widtha}p{\widthb}}\n";
    endinput = "\n\\endinput\n";
    newline = "\\tabularnewline";
  ];

  WriteString[ os, "\\begin{",environ,"}",
               If[ isTabular, cols ],
               If[ isTabular, tabhead, "\n" ] ];
  For[ j = 1, j < n, j++,
    {lhs, rhs} = tmp[[j]];
    If[ isTabular == True,
      WriteString[ os, "   \mx{ ", j, " } &  \mx{ ", TeXForm[rhs],
                   " }", If[ j != (n-1), newline, ""], "\n"];,
    (*else*)
      WriteString[ os, "  ",TeXForm[lhs]," & = ", TeXForm[rhs],
                   If[ j != (n-1), newline, ""], "\n"];
    ];
  ];
  WriteString[ os, "\\end{",environ,"}\n", endinput];
];
```

2.5.3 Tables of coefficients

21 ⟨*function to LaTeX material as coefficient tables* 21⟩≡ (19b)

```
LaTeXEqsTable[ eqs_, opts___Rule ] :=
  Module[{aj, bj, tmp, lst, rhsb, rhsc, tabhead, j, k, m,
          n = Length[ eqs[[2]] ]},
    pickupTeXOptions[ EquationsLaTeXed, opts ];
    pickupGenOptions[ Equations, opts];

    nlabel[os_, n_] := WriteString[ os, "%% n = ", n, "\n" ];

    os = myOpenWrite[ "-a", opts ];
    nlabel[os, n];

    (* send out Root objects first if appearing *)
    If[ Length[eqs] > 2,
      tabhead = TableHead[
      "  \\caption{Unique \\texttt{Root} objects $n=" <> ToString[n] <>
         "$.\\label{tab:c:" <>
         ToString[n] <> "}}\\\\\\n",
      "  \\caption[]{Algebraic numbers, $n=" <> ToString[n] <> "$.",
```

```
    "  \\parbox[c][20pt][c]{\\widtha}{\\centerline{$\\ell$}} & " <>
      "\\parbox[c][20pt][c]{\\widthb}{\\centerline{$c_{\\ell}$}}\\\\\n" ];
    texconsts[ Sort[ eqs[[3]], noLettersQ ], tabhead, os, opts ];
  ];

  (* send out intra-nodal stepsizes with some fancy rule sorting *)
  tabhead = TableHead[
    "  \\caption{Intra-nodal points $h_j$, $n="<> ToString[n] <>
       "$." <> "\\label{tab:h:" <> ToString[n] <> "}}\\\\\n",
    "  \\caption[]{Intra-nodal abscissa $h_j$, $n="<>
       ToString[n]<>"$.",
    "  \\parbox[c][20pt][c]{\\widtha}{\\centerline{$j$}} & " <>
      "\\parbox[c][20pt][c]{\\widthb}{\\centerline{$h_j$}}\\\\\n" ];
  texconsts[ Sort[ eqs[[1]], noLettersQ ], tabhead, os, opts ];
  Close[os];

  os = myOpenWrite[ "-b", opts ];
  nlabel[os, n];
  WriteString[ os, "\\begin{longtable}{p{\widtha}p{\widthb}}\n",
    TableHead[
      "  \\caption{Integral approximation $I_j$ for $n=" <>
         ToString[n] <> "$.\\label{tab:I:" <> ToString[n] <>
         "}}\\\\\n",
      "  \\caption[]{Integral approximation $I_j$, $n="<>
         ToString[n]<>"$.",
      "  \\parbox[c][20pt][c]{\\widtha}{" <>
         "\\centerline{$d_j,w_{jk}$}} &" <>
        "\\parbox[c][20pt][c]{\\widthb}{" <>
          "\\centerline{Expression}}\\\\\n"]];
  For[ j = 1, j <= n, j++,
    {aj, lst} = eqs[[2,j,{3,4}]];
    aj = Coefficient[ aj, sumcoefficient[[1]] ];
    WriteString[ os, "     \mx{ d_{", j ,"} } & \mx{ ",
                 TeXForm[ aj ]," } \\\\", "\n" ];

    For[ k = 0; tmp = lst[[All,1]]; m = Length[tmp], k < m, k++,
      WriteString[ os, "   \mx{ w_{", j,",",k, "} } & \mx{ ",
                   TeXForm[ tmp[[k+1]] ],
                   " } \\tabularnewline", "\n" ];
    ];
  ];
  WriteString[ os, "\\end{longtable}\n"];
  WriteString[ os, "\n\\endinput\n"];
  Close[os];
```

```
  ];
```

2.5.4 List of equations

23 ⟨*function to LaTeX material as standard equations* 23⟩≡ (19b)

```
LaTeXEqsFull[ eqs_, opts___Rule ] :=
  Module[ {tmp, lhs, rhsa, rhsb, rhsc, j, n},
    pickupTeXOptions[ EquationsLaTeXed, opts ];
    pickupGenOptions[ Equations, opts ];

    os = myOpenWrite[ "", opts ];
    WriteString[ os, "%% n = ", ToString[ Length[ eqs[[2]] ] ], "\n" ];

    (* send out Root objects first if appearing *)
    If[ Length[eqs] > 2,
      WriteString[ os, "\\begin{subequations}\n"];
      texconsts[ Sort[ eqs[[3]], noLettersQ ], "", os, opts ];
      WriteString[ os, "\\end{subequations}\n\n"];
    ];

    (* send out intra-nodal stepsizes with some fancy rule sorting *)
    WriteString[ os, "\\begin{subequations}\n"];
    texconsts[ Sort[ eqs[[1]], noLettersQ ], "", os, opts ];
    WriteString[ os, "\\end{subequations}\n\n"];

    WriteString[ os, "\\begin{subequations}\n"];
    tmp = eqs[[2]];
    n = Length[tmp];
    For[ j = 1, j <= n, j++,
      {lhs, rhsa} = tmp[[j,{1,2}]];
      WriteString[ os, "  \\begin{dmath}\n"];
      WriteString[ os, "  ", TeXForm[ lhs ], " & = \n" ];
      WriteString[ os, "    ", TeXForm[ rhsa ], " + \n"];
      {rhsb, rhsc} = tmp[[j,{3,4}]];
      WriteString[ os, "    ", TeXForm[ rhsb ], " \\left[ ",
                   TeXForm[ rhsc ], " \\right] \n",
                   "  \\end{dmath}\n"];
    ];
    WriteString[ os, "\\end{subequations}\n\n\\endinput\n"];
    If[ StringQ[ofn], Close[os] ];
  ];
```

2.5.5 LaTeX production public export

24a ⟨*publicly exported LaTeX output function* 24a⟩≡ (19b)

```
LaTeXeqs[ eqs_, opts___Rule ] :=
  Module[ {},
    pickupTeXOptions[ EquationsLaTeXed, opts ];

    If[ isTabular == True,
      LaTeXEqsTable[ eqs, opts ];,
    (*else*)
      LaTeXEqsFull[ eqs, opts ];
    ];
  ];
```

2.6 Root code chunks

The code chunks contained in this section represent the root chunks for both families of methods. In addition to these root chunks, a sample driver script for the family of first-order methods is provided that generates the methods, produces LaTeX output and stores the information in appropriate files.

2.6.1 Root chunk for family of first-order methods

24b ⟨`order1.mma` 24b⟩≡

  ⟨*package header:* $\mathcal{O}(1)$ 26b⟩
  ⟨*public exports to* `Global‘` *context:* $\mathcal{O}(1)$ 27b⟩
  ⟨*public exports to* `Global‘` *context: common* 28b⟩
  ⟨*begin* `Private` *context definitions* 30a⟩
    ⟨*options processing functions* 30b⟩
    ⟨*mathematical definitions: common* 7⟩
    ⟨*mathematical definitions:* $\mathcal{O}(1)$ 9b⟩
    ⟨*simplification engine: common:head* 11d⟩
    ⟨*simplification engine:* $\mathcal{O}(1)$ 15a⟩
    ⟨*simplification engine: common:tail* 15c⟩
    ⟨*utility functions* 16a⟩
    ⟨*LaTeX output functions* 19b⟩
  ⟨*package tail: common* 30c⟩

2.6.2 Root chunk for family of special form second-order methods

24c ⟨`order2s.mma` 24c⟩≡

  ⟨*package header:* $\mathcal{O}(2)s$ 27a⟩

⟨*public exports to* `Global‘` *context: common* 28b⟩
⟨*begin* `Private` *context definitions* 30a⟩
⟨*public exports to* `Global‘` *context:* $\mathcal{O}(2)s$ 28a⟩
  ⟨*options processing functions* 30b⟩
  ⟨*mathematical definitions: common* 7⟩
  ⟨*mathematical definitions:* $\mathcal{O}(2)s$ 10d⟩
  ⟨*simplification engine: common:head* 11d⟩
  ⟨*simplification engine:* $\mathcal{O}(2)s$ 15b⟩
  ⟨*simplification engine: common:tail* 15c⟩
  ⟨*utility functions* 16a⟩
  ⟨*LaTeX output functions* 19b⟩
⟨*package tail: common* 30c⟩

2.6.3 Sample driver script for family of first-order methods

As a sample, the driver below produces the methods for first-order differential equations for even degrees $2 \leq n \leq 16$. Once generated, the MATHEMATICA® form is stored to a file, then the LaTeX output is generated and saved to file. Here, the chosen output form is tables of coefficients.

To produce methods for second-order differential equations, simply alter the `Get` to load the appropriate file.

25a ⟨`generate.mma` 25a⟩≡

```
Get["order1.mma"];
For[ n = 2, n < 16, n+=2,
  tmp = Equations[ n ] // Timing;
  Put[ tmp, "n"<>ToString[n]<>".mma"];
  Print[ "time = ", tmp[[1]] ];
  LaTeXeqs[ tmp[[2]], OutputFilePrefix -> "n"<>ToString[n],
                      TeXTableForm->True ];
];
Print["Done."];
```

2.7 Miscellaneous chunks

25b ⟨*file header: common:head* 25b⟩≡ (26b 27a)

```
(* $Id: package.nw,v 1.32 2002/05/29 15:01:50 mitchejw Exp $ *)

(* :Title: *)

(* :Author: Jason Wm. Mitchell *)
```

26a ⟨*file header: common:tail* 26a⟩≡ (26b 27a)

```
(* :Context: Global` *)

(* :Package Version: 0.1 *)

(* :Copyright:
    Copyright (C) 2001 Jason Wm. Mitchell, All rights reserved.

    This software is released under the Perl Artistic License,
    see http://www.perl.com/pub/a/language/misc/Artistic.html

    This software is OSI Certified Open Source Software.
    OSI Certified is a certification mark of the Open Source Initiative.
*)

(* :History: *)

(* :Keywords:  Numerical integration, initial-value problem,
               Chebyshev methods, implicit methods,
               high-order methods
*)

(* :Mathematica Version: 3.x, 4.x *)

(* :Limitation(s): *)

(*====================================================================*)
```

26b ⟨*package header:* $\mathcal{O}(1)$ 26b⟩≡ (24b)

⟨*file header: common:head* 25b⟩

```
(* :Name: PanvoskyRichardsonOrder1 *)

(* :Summary:   PanvoskyRichardsonOrder1 is a package that generates
               equations for the numerical integration of initial
               value problems of the form
               y'(x) = f(y(x);x), y(0) = c1,
               using an implicit, intra-step Chebyshev interpolation.

               The equations are generated with exact coefficients
               based on the Chebyshev weights.  The user must
               implement the (implicit) iteration procedure;
               successive substitution.
```

```
*)

(* :Source: Richardson, D. L. and Panovsky, J., ``A Family of
            implicit Chebyshev methods for the numerical
            integration of first-order differential equations'',
            Proceedings of the AAS/AIAA Astrodynamics Specialists
            Conference, 1993.
*)

⟨file header: common:tail 26a⟩

BeginPackage["PanvoskyRichardsonOrder1`"];

ClearAll["PanovskyRichardsonOrder1`*"];
```

27a ⟨*package header:* $\mathcal{O}(2)s$ 27a⟩≡ (24c)

```
⟨file header: common:head 25b⟩
(* :Name: PanvoskyRichardsonOrder2S *)

(* :Summary:   PanvoskyRichardsonOrder2S is a package that generates
               equations for the numerical integration of initial
               value problems of the form
               y''(x) = f(y(x);t), y(0) = c1, y'(0) = c2,
               using an implicit, intra-step Chebyshev interpolation.

               The equations are generated with exact coefficients
               based on the Chebyshev weights.  The user must
               implement the (implicit) iteration procedure;
               successive substitution.

*)

(* :Source:  Panovsky, J. and Richardson, D. L., ``A Family of
             implicit Chebyshev methods for the numerical
             integration of second-order differential equations'',
             J. Comp. Appl. Math., 23, 1988, pp. 35--51.
*)

⟨file header: common:tail 26a⟩
BeginPackage["PanvoskyRichardsonOrder2S`"];

ClearAll["PanovskyRichardsonOrder2S`*"];
```

27b ⟨*public exports to* `Global‘` *context:* $\mathcal{O}(1)$ 27b⟩≡ (24b)

```
PanvoskyRichardsonOrder1::usage =
  "PanvoskyRichardsonOrder1 is a package that generates equations "
   "for the numerical integration of initial value problems of the "
   "form y'(x) = f(y(x);x), y(0) = c1, using an implicit, "
   "intra-step Chebyshev interpolation.\n\n"
   "The equations are generated with exact coefficients based on the "
   "Chebyshev weights.  The user must implement the (implicit) "
   "iteration procedure; successive substitution.";
Equations::usage =
  "Equations[ n, options... ] generates equations of order n for the "
   "numerical integration of systems of equations of the form "
   "y'(x) = f(y(x);x), y(0) = c1.";
```

28a ⟨*public exports to* `Global‘` *context:* $\mathcal{O}(2)s$ 28a⟩≡ (24c)

```
PanvoskyRichardsonOrder2s::usage =
  "PanvoskyRichardsonOrder2s is a package that generates equations "
   "for the numerical integration of initial value problems of the "
   "form y''(x) = f(y(x);x), y(0) = c1, y'(0) = c2, using an implicit, "
   "intra-step Chebyshev interpolation.\n\n"
   "The equations are generated with exact coefficients based on the "
   "Chebyshev weights.  The user must implement the (implicit) "
   "iteration procedure; successive substitution.";
Equations::usage =
  "Equations[ n, options... ] generates equations of order n for the "
   "numerical integration of systems of equations of the form "
   "y''(x) = f(y(x);x), y(0) = c1, y'(0) = c2.";
```

28b ⟨*public exports to* `Global‘` *context: common* 28b⟩≡ (24)

⟨*function descriptions: common* 28c⟩
⟨*function* `Equations` *option descriptions* 29a⟩
⟨*function* `LaTeXeqs` *option descriptions* 29b⟩
⟨*define function options and associations* 29c⟩

28c ⟨*function descriptions: common* 28c⟩≡ (28b)

```
LaTeXeqs::usage =
  "LaTeXeqs[ eqs, options... ] LaTeXs the output of Equations[n].";
IsDesiredForm::usage =
  "Predicate to determine if an expression is devoid of specific "
   "algebraic forms";
WLC::usage =
  "A leaf count function weighted against Root, Complex and (-1)^(n/m)...";
MakeSym::usage =
```

```
    "Routine to create symbols like x1, from tag (x) and counter."
  CosRepList::usage =
    "Replace Cos[n Pi/m] with irrational (algebraic) numbers.";
  GenEqsRaw::usage =
    "generate raw equation pieces";
```

29a ⟨*function* `Equations` *option descriptions* 29a⟩≡ (28b)

```
  IndependentVariable::usage =
    "Independent variable; default:  x.";
  ConstantsLabel::usage =
    "Label prefix for Root[] object constants; default:  c."
  ExactCoefficients::usage =
    "Exact or floating point coefficients switch; default:  True";
  NumericalPrecision::usage =
    "Number of decimal digits of precision for floating point "
     "coefficient representation; default:  16.";
```

29b ⟨*function* `LaTeXeqs` *option descriptions* 29b⟩≡ (28b)

```
  TeXTableForm::usage =
    "Boolean LaTeX option to put results in a table friendly form; "
     "default:  False."
  OutputFilePrefix::usage = "
    Output filename prefix; default:  Null."
  OutputStreamList::usage =
    "Stream list to write output; overridden by OutputFilePrefix; "
     "default: $Output";
```

The replacement rule `IndependentVariable->Global‘b` may seem strange since the independent variable discussed thus far has been x. However, MATHEMATICA® relies on a lexically sorted canonical form, so that $f(x+h)$ appears as `f[h+x]` because lexically `h > x`. To provide the preferred form while still subject to a lexical canonical from, `b` was arbitrarily selected as the independent variable so that $f(b+h)$ appears as `f[b+h]`. By calling `pickupGenOptions` were appropriate, `x` can be used in code while `b` with appearing in the output.

29c ⟨*define function options and associations* 29c⟩≡ (28b)

```
  ‘GenEqsOptionsList = { IndependentVariable->Global‘b,
                         DependentVariable->Global‘y,
                         DerivativeFunction->Global‘f,
                         StepSize->Global‘h,
                         ConstantsLabel->Global‘c,
                         ExactCoefficients->True,
```

```
                      NumericalPrecision->16 };

`TeXEqsOptionsList = { TeXTableForm->False,
                       OutputFilePrefix->Null,
                       OutputStreamList->$Output };

Options[Equations] = `GenEqsOptionsList;

Options[EquationsLaTeXed] = `TeXEqsOptionsList;
```

30a ⟨*begin* `Private` *context definitions* 30a⟩≡ (24)

```
Begin["`Private`"];

(* explicitly declare all local variables to avoid Global`* collisions *)
{ `x, `y, `f, `h, `c, `protected, `exactp, `prec, `ofn, `osl };
```

30b ⟨*options processing functions* 30b⟩≡ (24)

```
pickupGenOptions[ from_, opts___Rule ] :=
   Module[ {},
     x = IndependentVariable    /. {opts} /. Options[ from ];
     y = DependentVariable      /. {opts} /. Options[ from ];
     f = DerivativeFunction     /. {opts} /. Options[ from ];
     h = StepSize               /. {opts} /. Options[ from ];
     c = ConstantsLabel         /. {opts} /. Options[ from ];
     exactp = ExactCoefficients /. {opts} /. Options[ from ];
     prec = NumericalPrecision  /. {opts} /. Options[ from ];
   ];

pickupTeXOptions[ from_, opts___Rule ] :=
   Module[ {},
     isTabular = TeXTableForm /. {opts} /. Options[ from ];
     ofn = OutputFilePrefix   /. {opts} /. Options[ from ];
     osl = OutputStreamList   /. {opts} /. Options[ from ];
   ];
```

30c ⟨*package tail: common* 30c⟩≡ (24)

```
(* restore protections *)
Protect[ Evaluate[protected] ];

End[];                  (* end of private context *)
Protect[ ];             (* protect exported symbols *)
EndPackage[];           (* end of package *)
```

Chapter 3

Implementing an Integrator for First-Order Systems

3.1 Introduction

Implementing a numerical integrator for first-order systems for a given degree n is straight forward. In the following, an integrator of degree $n = 8$ is constructed using the exact coefficients generated by the method described in Chapter 2 and computed numerically to double-precision.

3.2 Basic Requirements

To begin, it is necessary to select a programming language and compiler. For this example, C++ is the programming language of implementation. Since certain aspects of the Standard Library are specified incorrectly in the 1998 ISO standard (ISO/IEC 14882), a functional compiler must necessarily incorporate defect resolutions produced by the ISO/IEC standardization working group[1]. A compiler that addresses this issue and provides a functional Standard Library and Standard Template Library (STL) is the C++ compiler found in the GNU Compiler Collection, `gcc-3.1.0`.

With the programming language and compiler selected, we identify the highest level components necessary to implement an integrator of degree $n = 8$. From the previous chapters, we will need the associated exact coefficients computed to double-precision, a fourth-order Runge-Kutta guesser, and an iteration by continuous substitution similar to Algorithm 1.1.

As a final basic requirement, we must choose the data structure needed to manipulate the array data that will arise. Specifically, we would like a templated, fixed-size container that provides amortized constant time access, is not bounds checked, behaves like a mathematical vector, is optimized for speed, is amenable to the BLAS *slice*[2] abstraction, and interacts with STL algorithms. With a complete Standard

[1] see the JTC1/SC22/WG21 web page at `http://std.dkuug.dk/jtc1/sc22/wg21/`.

[2] see the Basic Linear Algebra Subprograms library at `http://www.netlib.org/blas/`.

Library and STL, there are many options available. Fortunately, a combination of the Standard Library container `valarray` and `slice` meets all of these requirements individually. Unfortunately, the combined requirement of interaction between slices and STL algorithms is not met by default because slices alone do not meet STL iterator requirements. This can be remedied with the addition of a `slice_iterator` class.

3.3 Root chunk: `slice_iterator`

The `slice_iterator` developed here is based largely on the *Slice_iter* class presented in Stroustrup [9, § 22.4.5, p. 670]. Several alterations have been made to that original design and will be discussed as they appear.

32a ⟨`slice_iterator` 32a⟩≡

```
⟨si: header 40c⟩
#ifndef PANOVSKY_RICHARDSON_SLICE_ITERATOR
#define PANOVSKY_RICHARDSON_SLICE_ITERATOR
⟨si: includes 40b⟩

namespace pr_cheby
{
  ⟨si: class decl 32b⟩
  ⟨si: member defs 36a⟩
}
#endif //PANOVSKY_RICHARDSON_SLICE_ITERATOR
```

In the case of `slice_iterator`, as with all the supporting code presented here, we choose to place all code in the header file for `inline` convenience.

3.3.1 Class declaration

32b ⟨*si: class decl* 32b⟩≡ (32a)

```
template <typename T>
class slice_iterator
{
  public:
    ⟨si: forward iterator typedef decls 33c⟩
    ⟨si: constructor decls 34a⟩
    ⟨si: forward iterator interface decls 34b⟩
    ⟨si: access operator decls 35a⟩
    ⟨si: comparison & ordering decls 35b⟩
  private:
    ⟨si: private data 33a⟩
    ⟨si: private function decls 33b⟩
```

```
};
```

⟨*si: global comparison & order decls* 35c⟩

To help prevent name clashes with other existing libraries, we introduce the `namespace pr_cheby`. All remaining code to support the integrator in this example will be introduced in this `namespace`.

3.3.1.1 Private data

The implementation of `slice_iterator` will require a certain amount of aggregated data to represent its internal state. Since we are building the iterator to work with `valarray`, we will need a pointer to the desired instance. Also, we will need the desired `slice` view, as well as an index indicating the current element.

33a ⟨*si: private data* 33a⟩≡ (32b)

```
std::valarray<T>* v_;
std::slice s_;
size_t cur_;
```

Since each data type is `private`, the identifiers are suffixed with an underscore to draw attention to their access membership. This notation will be used in all the presented in this example.

3.3.1.2 Private functions

The `ref` member provides a mechanism to calculate the slice element to be accessed in `v_`.

33b ⟨*si: private function decls* 33b⟩≡ (32b)

```
T& ref( size_t k );
const T& ref( size_t k ) const;
```

A `const`-qualified version of `ref` is provided.

3.3.1.3 Forward iterator declarations

Since the purpose of `slice_iterator` is to allow `slice` interaction with STL algorithms and provide element access, it must meet some minimal iterator interface. A minimal iterator interface for our needs is the `forward_iterator`.

33c ⟨*si: forward iterator* `typedef` *decls* 33c⟩≡ (32b)

```
typedef typename std::forward_iterator_tag iterator_category;
typedef T value_type;
typedef typename std::ptrdiff_t difference_type;
typedef T* pointer;
```

```
typedef T& reference;
```

The `iterator_category` is declared as a forward iterator, and the remaining declarations are provided as required by the `iterator_traits` template class, Stroustrup [9, § 19.2.2].

3.3.1.4 Constructors

A default constructor is supplied explicitly so that arrays of empty `slice`s may be constructed without difficulty. The second constructor below creates a specific instance of a `slice_iterator` given a `valarray` and its `slice`.

Since the `slice_iterator` is very simple with respect to data (§ 3.3.1.1), no copy constructor or assignment operator is explicitly defined. The default compiler-supplied behaviour will be sufficient for copy and assignment.

34a ⟨*si: constructor decls* 34a⟩≡ (32b)
```
slice_iterator();
slice_iterator( std::valarray<T>* v, std::slice s );
```

3.3.1.5 Forward iterator interface

For the case of STL iterators, the member methods `begin()` and `end()` provide the interface that specifies the range of an operation over a container. These members return the first and one-past-the-end iterators, respectively. Stroustrup [9, p. 670] seems to consider only the case when the iterator is initially constructed and so does not provide a `begin()` member. Since we anticipate that we will want to repeatedly reuse existing iterator instances, a `begin()` member is supplied to avoid the additional overhead of unnecessarily recreating iterators to obtain a starting iterator element.

34b ⟨*si: forward iterator interface decls* 34b⟩≡ (32b) 34c ▷
```
slice_iterator begin();
slice_iterator end();
```

A forward iterator does not define any new expressions beyond the input iterator category, but some of the input iterator restrictions are relaxed so that only two pre- and post-increment operators need to be provided, i.e. `(void)i++` is dropped.

34c ⟨*si: forward iterator interface decls* 34b⟩+≡ (32b) ◁ 34b
```
slice_iterator& operator++();
slice_iterator operator++(int);
```

3.3.1.6 Access operators

To permit element access, appropriate operators are declared. The `operator*()` is provided for use with STL algorithms requiring sequential access, and `operator[]()` is provided for random access and to make `slice_iterator` feel more like an array, unlike `std::slice_array`.

35a ⟨*si: access operator decls* 35a⟩≡ (32b)

```
T& operator*();
const T& operator*() const;

T& operator[](size_t k);
const T& operator[](size_t k) const;
```

For each of the element access operators above, a `const`-qualified version is also declared. To minimize overhead, these `const`-qualified access operators return a reference to a constant element rather than a copy of the element accessed.

With regard to random access efficiency, the `operator[]()`, the input index of type `size_t` should not be bounds checked.

3.3.1.7 Comparison functions

To facilitate comparison of `slice_iterator` instances, it is preferable to provide a few public member functions that accomplish this task rather than declare a large block of `friend` functions. Since these members do not alter state, they are appropriately `const`-qualified.

35b ⟨*si: comparison & ordering decls* 35b⟩≡ (32b)

```
bool slice_equal( const slice_iterator<T>& in ) const;
bool is_equal( const slice_iterator<T>& in ) const;
bool is_less_than( const slice_iterator<T>& in ) const;
```

The member `slice_equal` should determine if the `slice` contained within `*this` is equivalent to the `slice` contained in the input `in`.

3.3.1.8 Global comparison and order functions

35c ⟨*si: global comparison & order decls* 35c⟩≡ (32b)

```
template <typename T> inline
bool operator==(const slice_iterator<T>& a, const slice_iterator<T>& b);

template <typename T> inline
bool operator!=(const slice_iterator<T>& a, const slice_iterator<T>& b);

template <typename T> inline
```

```
bool operator<(const slice_iterator<T>& a, const slice_iterator<T>& b);

template <typename T> inline
bool operator>(const slice_iterator<T>& a, const slice_iterator<T>& b);
```

These global operators are required by the algorithms found in the STL. Since `public` member functions are provided for comparison and ordering, these functions will simply be expanded `inline` as forwarding functions. Consequently, their call overhead will be eliminated at compile-time.

3.3.2 Member definitions

36a ⟨*si: member defs* 36a⟩≡ (32a)

⟨*si: private function defs* 36b⟩
⟨*si: constructor defs* 36d⟩
⟨*si: forward iterator interface defs* 37b⟩
⟨*si: access operator defs* 38a⟩
⟨*si: comparison & ordering defs* 39a⟩
⟨*si: global comparison & order defs* 40a⟩

3.3.2.1 Private functions

Computing the kth element of a `slice s` is straightforward, and defined below.

36b ⟨*si: private function defs* 36b⟩≡ (36a) 36c ▷

```
template <typename T> inline
T& slice_iterator<T>::ref( size_t k )
{ return( (*v_)[s_.start()+k*s_.stride()] ); }
```

The `const`-qualified `ref` is defined as

36c ⟨*si: private function defs* 36b⟩+≡ (36a) ◁ 36b

```
template <typename T> inline
const T& slice_iterator<T>::ref( size_t k ) const
{ return( (*v_)[s_.start()+k*s_.stride()] ); }
```

3.3.2.2 Constructors

In terms of constructors, we must be able to easily construct arrays of uninitialized `slice_iterator` instances, as well as instances that refer to a specific `slice` of a `valarray`.

The default constructor is

36d ⟨*si: constructor defs* 36d⟩≡ (36a) 37a ▷

```
template <typename T> inline
slice_iterator<T>::slice_iterator() : v_(0), cur_(0) {}
```

As defined, this constructor simply zeroes `v_` and `cur_`, indicating that the iterator is empty, allowing the easy construction of arrays of `slice_iterator`s.

To create a `slice_iterator` for a `slice` of a `valarray`, we define the following constructor.

37a ⟨*si: constructor defs* 36d⟩+≡ (36a) ◁ 36d

```
template <typename T> inline
slice_iterator<T>::slice_iterator( std::valarray<T>* v, std::slice s )
  : v_(v), s_(s), cur_(0) {}
```

Copy constructor, destructor, and assignment For each of these cases, the default, compiler-generated definitions are used.

3.3.2.3 Forward iterator interface

The implementation of the `begin()` and `end()` members is quite simple. A copy of `*this` is made and `cur_` is set either to zero or `s_.size()`, respectively.

37b ⟨*si: forward iterator interface defs* 37b⟩≡ (36a) 37c ▷

```
template <typename T> inline
slice_iterator<T> slice_iterator<T>::begin()
{
  slice_iterator<T> tmp = *this;
  tmp.cur_ = 0;
  return( tmp );
}

template <typename T> inline
slice_iterator<T> slice_iterator<T>::end()
{
  slice_iterator<T> tmp = *this;
  tmp.cur_ = s_.size();
  return( tmp );
}
```

To complete the iterator interface, we define the pre- and post-increment operators.

37c ⟨*si: forward iterator interface defs* 37b⟩+≡ (36a) ◁37b

```
template <typename T> inline
slice_iterator<T>& slice_iterator<T>::operator++()
{
  cur_++;
  return( *this );
}

template <typename T> inline
slice_iterator<T> slice_iterator<T>::operator++(int)
{
  slice_iterator<T> tmp = *this;
  cur_++;
  return( tmp );
}
```

3.3.2.4 Access operators

Element access is provided by `operator*()` and `operator[]()` and their `const`-qualified equivalents. These member methods are implemented via the previously defined `ref` member. For the `operator*()`, `ref` is called using the current index, `cur_`.

38a ⟨*si: access operator defs* 38a⟩≡ (36a) 38b▷

```
template <typename T> inline
T& slice_iterator<T>::operator*()
{ return( ref(cur_) ); }

template <typename T> inline
const T& slice_iterator<T>::operator*() const
{ return( ref(cur_) ); }
```

For random element access based on an input index, `operator[]()` is defined. It simply calls `ref` with the requested index.

38b ⟨*si: access operator defs* 38a⟩+≡ (36a) ◁38a

```
template <typename T> inline
T& slice_iterator<T>::operator[](size_t k)
{ return( ref(k) ); }

template <typename T> inline
const T& slice_iterator<T>::operator[](size_t k) const
{ return( ref(k) ); }
```

For each operator defined here, a `const`-qualified version is also defined.

As previously discussed in § 3.3.1.6, the access operators defined here do not check bounds on any element access request.

3.3.2.5 Comparison & ordering

To avoid the use of `friend` functions when providing definitions for the global comparison and ordering operators needed by STL algorithms, we choose to provide several `public` members to perform these tasks. Then, the global functions for comparison and ordering will then simply forward to these `public` members.

Equivalence between two `slice_iterator` instances will be established if their `slice`, current index, and container pointer all compare favorably. The member `slice_equal` checks that the `slice`s are equivalent.

39a ⟨*si: comparison & ordering defs* 39a⟩≡ (36a) 39b ▷

```
template <typename T> inline
bool slice_iterator<T>::slice_equal( const slice_iterator<T>& in ) const
{ return( s_.start()==in.s_.start() && s_.stride()==in.s_.stride() ); }
```

Two `slice_iterator`s are fully compared with the member method `is_equal`.

39b ⟨*si: comparison & ordering defs* 39a⟩+≡ (36a) ◁ 39a 39c ▷

```
template <typename T> inline
bool slice_iterator<T>::is_equal( const slice_iterator<T>& in ) const
{ return( v_==in.v_ && cur_==in.cur_ && slice_equal(in) ); }
```

The above definition differs from that of Stroustrup [9, § 22.4.5, p. 670] by the additional comparison of `v_`. The rational for this is straight forward. Stroustrup considers only the equivalence of the data view provided by the `slice_iterator`. However, it would seem that both the view and the data should compare favorably before equivalence between two `slice_iterator`s is established.

Ordering is accomplished with the definition of `is_less_than`.

39c ⟨*si: comparison & ordering defs* 39a⟩+≡ (36a) ◁ 39b

```
template <typename T> inline
bool slice_iterator<T>::is_less_than( const slice_iterator<T>& in ) const
{ return( v_==in.v_ && slice_equal(in) && cur_<in.cur_ ); }
```

In a similar fashion as with `is_equal`, we only consider ordering to be meaningful if the data and the `slice` are equivalent.

Each of the member methods defined here are `const`-qualified since their operations do not change the state of a `slice_iterator` instance.

3.3.2.6 Global comparison & ordering

As mentioned in the previous section, the global functions used by STL algorithms for comparison and ordering are implemented as forwarding functions to their counterpart defined in § 3.3.2.5.

40a ⟨*si: global comparison & order defs* 40a⟩≡ (36a)

```
template <typename T> inline
bool operator==( const slice_iterator<T>& a, const slice_iterator<T>& b )
{ return( a.is_equal(b) ); }

template <typename T> inline
bool operator!=( const slice_iterator<T>& a, const slice_iterator<T>& b )
{ return( !(a==b) ); }

template <typename T> inline
bool operator<( const slice_iterator<T>& a, const slice_iterator<T>& b )
{ return( a.is_less_than(b) ); }

template <typename T> inline
bool operator>( const slice_iterator<T>& a, const slice_iterator<T>& b )
{ return( !(a<b) ); }
```

3.3.2.7 Header & includes

The `slice_iterator` translation unit is completed with the following include information and header.

40b ⟨*si: includes* 40b⟩≡ (32a)

```
#include <cstddef>
#include <iterator>
#include <valarray>
```

40c ⟨*si: header* 40c⟩≡ (32a)

```
//===========================================================================
//   Project:  DTIC technical report example integrator implementation
//
// Namespace:  pr_cheby
//
//   Purpose:  slice iterator used by Chebyshev integrator of degree n = 8.
//
// $RCSfile: slice-iterator.nw,v $
// $Revision: 1.17 $
//
```

```
// $State: Exp $
// $Locker:  $
//
// $Source: /home/mitchejw/work/research/tex/dtic/tr/RCS/slice-iterator.nw,v $
//
// Comments:  0. Since valarray<> is used, slice_iterators are necessary to
//               use STL functions.
//
//            1. See Stroustrup, "The C++ Programming Language, Special Ed."
//               for more details
//
//   Orig. Author:  Jason Wm. Mitchell
//                  US Air Force Research Laboratory
//                  Air Vehicles Directorate, AFRL/VACA
//                  2210 Eighth Street, Bldg. 146, Rm. 305
//                  Wright-Patterson AFB, OH  45433
//         E-Mail:  jason+dtic_report2002@maiar.org
//     Orig. Date:  02-Feb-2002
//   Last Changed:  $Date: 2002/05/29 15:10:39 $
//
// Copyright (C) 2002 Jason Wm. Mitchell, All Rights Reserved.
// Restrictions:  released under the Perl Artistic license.
// Security:      unclassified
//
// This software is OSI Certified Open Source Software.
// OSI Certified is a certification mark of the Open Source Initiative.
//==============================================================================
```

3.4 Root chunk: `default_guesser`

As previously suggested initially in § 1.2.3 and identified as a basic requirement in § 3.2, a fourth-order Runge-Kutta scheme will be used to jump-start the process described by Algorithm 1.1.

In terms of design concepts, it is useful to view the guess generator, and integrators in general, as an operator transforming an input vector to some output vector. As an implementation, treating this operator like a function is very convenient. For example, let the operation of guess generation be represented by the function g. Then, for given a first-order system of equations, stepsize h, initial independent variable x_0, and initial state $\mathbf{y}_0$, the next state $\mathbf{y}$ is simply $g : \mathbf{y}_0 \mapsto \mathbf{y}$. Since a fourth-order Runge-Kutta method requires storage of intermediate derivatives, some contextual data is associated with the operator and we may think of it as a closure. We implement this by providing an `operator()()` member.

As with the previous code, we introduce all declarations and definitions in the

pr_cheby namespace.

42a ⟨default_guesser 42a⟩≡

```
⟨dg: header 46b⟩
#ifndef PANOVSKY_RICHARDSON_DEFAULT_GUESSER
#define PANOVSKY_RICHARDSON_DEFAULT_GUESSER
⟨dg: includes 46a⟩
namespace pr_cheby
{
  using namespace std;

  ⟨dg: class decl 42b⟩

  ⟨dg: member defs 44a⟩
}
#endif // PANOVSKY_RICHARDSON_DEFAULT_GUESSER
```

3.4.1 Class declaration

The default_guesser object is parameterized by the scalar type T.

42b ⟨*dg: class decl* 42b⟩≡ (42a)

```
template <typename T >
class default_guesser
{
 public:
  ⟨dg: typedef decls 43b⟩
  ⟨dg: constructors decls 43c⟩
  ⟨dg: member-template operator()() decl 43d⟩
 private:
  ⟨dg: private data decls 42c⟩
  ⟨dg: private function decls 43a⟩
};
```

3.4.1.1 Private data

The internal data representation is relatively small. We need to know that the Runge-Kutta method is fourth-order n_, the number of first-order equations m_, and we must have storage for the fully explicit intermediate derivative values k_.

42c ⟨*dg: private data decls* 42c⟩≡ (42b)

```
static const size_t n_ = 4;
const size_t m_;
mutable valarray<valarray<T> > k_;
```

The method order is fixed, and so `n_` is `static`, `const`, of integral type, and thus defined in the class declaration. Also, the number of first-order equations `m_` will be considered to be constant. The intermediate derivatives `k_` are stored in a `valarray` of `valarrays`. The `mutable` qualification of `k_` is necessary to permit constant reference invocations. For this case, we choose the emphasize the stateless mapping operator behaviour of the guesser, rather than an object requiring intermediate states to complete its task. This may seem at odds with the general design philosophy of using constant objects, but its use here is well understood and documented.

3.4.1.2 Private functions

The only `private` function declaration given here is used to ban default construction. We prefer to disallow user attempts to create uninitialized `default_guesser` instances. This restriction is for clarity and programming convenience.

43a ⟨*dg: private function decls* 43a⟩≡ (42b)
```
default_guesser();
```

Since we only seek to generate compile-time errors with this mechanism, the function need not be defined.

3.4.1.3 `typedef` declarations

The declaration of `class default_guesser` is parameterized by type `T` which is captured in a `typedef` declaration.

43b ⟨*dg:* `typedef` *decls* 43b⟩≡ (42b)
```
typedef T value_type;
```

3.4.1.4 Constructors

The only input required to construct an instance of `default_guesser` is the number of first-order equations `m` that are necessary. A copy constructor is also provided[3].

43c ⟨*dg: constructors decls* 43c⟩≡ (42b)
```
default_guesser( const size_t& m );
default_guesser( const default_guesser& );
```

3.4.1.5 Member-template `operator()()`

The function-like behaviour is provided by the member-template `operator()()`.

[3]See § 3.4.1.2 for the discussion regarding the lack of a public default constructor.

43d ⟨*dg: member-template* `operator()()` *decl* 43d⟩≡ (42b)

```
template <typename V1, typename V2, typename E >
void operator()( const T& x, const V1& y,
                 const T& h, const E& eqs,
                 V2& y_new ) const;
```

Choosing a member-template provides the necessary type flexibility while retaining type-safety. While `default_guesser` is parameterized by `T` already, this member-template adds the `V1`, `V2`, and `E` types. The types `V1` and `V2` are intended to represent vector-like quantities and that provide an `operator[]()` access method. The type `E` represents the system of first-order equations to be evaluated.

The argument signature for the member-template expects the current independent variable `x`, the current state `y`, the fixed stepsize `h`, the first-order equations `eqs`. These arguments are considered inputs and are `const`-qualified. The only output argument is `y_new` of type `V2` which is the estimate of $\mathbf{y}(x+h)$.

The return type of the member-template is `void` because the writable storage for the output `y_new` is passed into the method in the argument list. In addition, the method is `const`-qualified.

One additional user enforced requirement here is that the scalar types of `V1` and `V2` be of type `T`. This is not an unreasonable expectation and is a most likely scenario.

3.4.2 Member definitions

44a ⟨*dg: member defs* 44a⟩≡ (42a)
  ⟨*dg: constructor defs* 44b⟩
  ⟨*dg: member-template* `operator()()` *def* 45b⟩

3.4.2.1 Constructors

Construction of the appropriate temporary storage `k_` and retention of the number first-order equations `m_` is performed by the user constructor below.

44b ⟨*dg: constructor defs* 44b⟩≡ (44a) 45a ▷

```
template <typename T> inline
default_guesser<T>::default_guesser( const size_t& m )
  : m_(m), k_( valarray<T>(m), n_ )
{
  assert( m_ > 0 );
}
```

The constructor above list assigns `m_` and uses `valarray`'s array constructor to allocate four `valarrays` of length `m`. The `assert` macro is used, ex post facto, to ensure that `m` is strictly positive. A `sentry` instance could be a used to enforce this requirement before construction of `k_`, however the `assert` macro is sufficient for

this example. No attempt is made to verify that the requested memory was actually allocated. In addition, any exceptions thrown from custom allocators are ignored.

The copy constructor definition attempts to make sure the appropriate objects are constructed and copied.

45a ⟨*dg: constructor defs* 44b⟩+≡ (44a) ◁44b

```
template <typename T> inline
default_guesser<T>::default_guesser( const default_guesser<T>& g )
  : m_(g.m_), k_( valarray<T>(g.m_), n_ ) {}
```

It is worth noting that the values found in `g.k_` are not copied because they are always computed when needed.

3.4.2.2 Member-template `operator()()`

The Runge-Kutta algorithm presented here is the original classical generalization of Simpson's rule[4]. The following implementation of this algorithm attempts to minimize necessary storage by using the output `y_new` as additional intermediate storage. The intermediate derivative values are stored in the previously allocated `k_`.

45b ⟨*dg: member-template* `operator()()` *def* 45b⟩≡ (44a)

```
template <typename T>
template <typename V1, typename V2, typename E > inline
void default_guesser<T>::operator()( const T& x, const V1& y,
                                     const T& h, const E& eqs,
                                     V2& y_new ) const
{
  const T h_on_2 = h / T(2);

  eqs( x, y, k_[0] );
  y_new = y + h_on_2 * k_[0];

  eqs( x + h_on_2, y_new, k_[1] );
  y_new = y + h_on_2 * k_[1];

  eqs( x + h_on_2, y_new, k_[2] );
  y_new = y + h * k_[2];

  eqs( x + h, y_new, k_[3] );
  y_new = y + h*( k_[0] + T(2)*(k_[1]+k_[2]) + k_[3] ) / T(6);
}
```

To allow for the possibility that the stepsize `h` given to the algorithm may change, it is passed in at each method call. But, like all of the method arguments, it is passed

[4] For more details on the development of this algorithm, see Atkinson [1, p.423].

by reference to prevent unnecessary copying during each method call. This *hand* optimization is a concern because this method will be frequently invoked.

The algorithm as described here clearly sets the interface requirements for invoking the system of first-order equations. They must behave in a function-like fashion, and accept the arguments of independent variable `x`, a state value `y` or `y_new`, and output storage for the derivative values `k_[`j`]`. These arguments must be of type `const T&`, `const V1&` or `const V2&`, and `valarray<T>&`, respectively. As previously mentioned, specifying references will avoid unnecessary copying. This will be important for the derivatives because we expect them to be the most evaluated function over the integration.

3.5 Includes & header

The include files necessary are `cstddef`, `cassert`, and `valarray`.

46a ⟨*dg: includes* 46a⟩≡ (42a)

```
#include <cstddef>
#include <cassert>
#include <valarray>
```

46b ⟨*dg: header* 46b⟩≡ (42a)

```
//=============================================================================
//   Project:  DTIC technical report example integrator implementation.
//
// Namespace:  pr_cheby
//
//   Purpose:  Default guess generator for intra-nodal points
//
// $RCSfile: default-guesser.nw,v $
// $Revision: 1.19 $
//
// $State: Exp $
// $Locker:  $
//
// $Source: /home/mitchejw/work/research/tex/dtic/tr/RCS/default-guesser.nw,v $
//
// Comments:  0. This represents the wrapping for the intra-nodal
//                guess generator.  The default guesser is a classical
//                Runge-Kutta order four (4) guesser; simple coefficients.
//
//            1. Check operator() for the interface to y(x)\mapsto y(x+h)
//
//   Orig. Author:  Jason Wm. Mitchell
//                  US Air Force Research Laboratory
```

```
//                    Air Vehicles Directorate, AFRL/VACA
//                    2210 Eighth Street, Bldg. 146, Rm. 305
//                    Wright-Patterson AFB, OH  45433  USA
//           E-Mail:  jason+dtic_report2002@maiar.org
//       Orig. Date:  02-Feb-2002
//     Last Changed:  $Date: 2002/05/29 14:56:19 $
//
// Copyright (C) 2002 Jason Wm. Mitchell, All Rights Reserved.
// Restrictions:  released under Perl Artistic license
// Security:      unclassified
//
// This software is OSI Certified Open Source Software.
// OSI Certified is a certification mark of the Open Source Initiative.
//==============================================================================
```

3.6 Root chunk: `pr_constants`

To allow the integrator access to the MATHEMATICA® generated coefficients for degree $n = 8$ in a type abstract manner, we introduce these coefficients in a *traits* template[5]. The double-precision version of these coefficients, accurate to twenty decimal places, are provided as the traits template *specialized* for type `double`.

In addition to the integrator coefficients, the traits template should include any other precision related constants that the integrator implementation may require. By isolating the floating-point constants in this way, we ensure that different precision floating-point constants can be used without altering integration algorithm. In terms of implementation, only the traits template specialized for the new type would need to be specified. For example, a `long double` specialization could be provided for the traits template rather than the `double`. Then, by choosing `long double` as the scalar type for all operations, the integrator would, at compile-time, deduce that it should use the traits template specialized for `long double`, thus making algorithm modification unnecessary.

47 ⟨`pr_constants` 47⟩≡

```
⟨prc: header 53b⟩
#ifndef PANOVSKY_RICHARDSON_CHEBYSHEV_8TH_DEGREE_CONSTANTS
#define PANOVSKY_RICHARDSON_CHEBYSHEV_8TH_DEGREE_CONSTANTS
⟨prc: includes 53a⟩

namespace pr_cheby
{
  using namespace std;
```

[5] *Traits* template classes conveniently aggregate type dependent template parameters.

```
  ⟨prc: traits decl 48a⟩
  ⟨prc: double specialization 49a⟩
}
#endif // PANOVSKY_RICHARDSON_CHEBYSHEV_8TH_DEGREE_CONSTANTS
```

3.6.1 Traits template declaration

48a ⟨*prc: traits decl* 48a⟩≡ (47)

```
template <typename T>
struct constants
{
  static const bool is_specialized = false;

  ⟨prc: integral type decls 48b⟩
  ⟨prc: floating type decls 48c⟩
};
```

The `is_specialized` member provides a compile time mechanism to verify that the appropriate specialization is provided and available.

3.6.1.1 Integral type members

In addition to the integrator constants, we include constants for the maximum number of iterations permitted on an interval and the iteration residual tolerance. Also, since it may be useful to relax the residual tolerance, we include a maximum number of unconverged iterations and a multiplicative growth factor for the residual tolerance.

48b ⟨*prc: integral type decls* 48b⟩≡ (48a)

```
typedef size_t size_type;
static const size_t n = 8;
static const size_t maximum_iterations  = 0;
static const size_t maximum_unconverged_iterations = 0;
```

Selection of the values related to the maximum number of iterations should be based on some experimentation with the desired problem.

3.6.1.2 Floating type members

The interpolating weights `w` are stored as an arrays of arrays of type `T`. To facilitate accessing rows of weights for inner product operations, a `typedef` declaration is provided. Also, a `typedef` declarations is provided for the parameterizing type.

Along with the interpolation weights, two other sets of constants are useful to precompute for the iteration algorithm: interior ξ_j and $\Delta\xi_j = \xi_j - \xi_{j-1}$ for $0 < \xi \leq n+1$.

48c ⟨*prc: floating type decls* 48c⟩≡ (48a)
```
typedef T value_type;
typedef T const (* const array_type)[ n+1 ];
static const T epsilon;
static const T epsilon_grow_factor;
static const T  xi[ n-1 ];
static const T dxi[ n ];
static const T   w[][ n+1 ];
```

In the declaration of `array_type`, we have deviated from the convenience use of the `const`-qualifier and moved to the explicit use which binds to the left.

3.6.2 Explicit specialization for `double`

The implementation precision chosen is `double` for the Intel 32-bit platform.

49a ⟨*prc:* `double` *specialization* 49a⟩≡ (47)
```
template <>
struct constants<double>
{
  static const bool is_specialized = true;

  ⟨prc dbl: integral type decls 49b⟩
  ⟨prc dbl: floating type decls 49c⟩
};

⟨prc dbl: floating type defs 50a⟩
```

The member `is_specialized` is set to indicate explicit specialization.

3.6.2.1 Specialized integral type members

49b ⟨*prc dbl: integral type decls* 49b⟩≡ (49a)
```
typedef size_t size_type;
static const size_t n = 8;
static const size_t maximum_iterations  = 16;
static const size_t maximum_unconverged_iterations = 6;
```

The values related to the maximum number of iterations have been chosen with the problem of an harmonic oscillator in mind.

3.6.2.2 Specialized floating type members

49c ⟨*prc dbl: floating type decls* 49c⟩≡ (49a)
```
typedef double value_type;
```

```
typedef double const (* const array_type)[ n+1 ];
static const double epsilon;
static const double epsilon_grow_factor;
static const double  xi[ n-1 ];
static const double dxi[ n ];
static const double   w[][ n+1 ];
```

3.6.2.3 Specialized constant definitions

For the target platform, double-precision means approximately sixteen decimal places of accuracy, and so the floating-point constants given here follow the standard C++ library implementation convention of providing approximately twenty decimal places in the definitions.

Tolerance related constants: The iteration residual tolerance is chosen to be to be twice the *unit round* provided by `numeric_limits<double>::epsilon()`. If the decision is made to relax the tolerance requirement by using a multiplier, that factor is taken to be 2. These particular values are chosen only after relatively little experimentation for a specific and well understood problem and should not be considered a uniform choice for all problems.

50a ⟨*prc dbl: floating type defs* 50a⟩≡ (49a) 50b ▷

```
const double constants<double>::epsilon =
numeric_limits<double>::epsilon() * 2;

const double constants<double>::epsilon_grow_factor = 2.0;
```

Interior intra-nodal points: The interior ξ_j locations are computed exactly and stored. The end-points are excluded to reduce rounding errors in the algorithm. As a result of this, evaluations at the end-points must be handled individually.

50b ⟨*prc dbl: floating type defs* 50a⟩+≡ (49a) ◁ 50a 51a ▷

```
template <>
const double constants<double>::xi[] =
{
  3.8060233744356621936e-02,
  1.4644660940672623780e-01,
  3.0865828381745511414e-01,
  5.0000000000000000000e-01,
  6.9134171618254488586e-01,
  8.5355339059327376220e-01,
  9.6193976625564337806e-01,
```

```
};
```

Intra-nodal spacing: Algorithm 1.1 requires that guesses are generated at the intra-nodal points. Since the intra-nodal spacing is not fixed, $\Delta\xi_j$ is computed and stored for use by the guess generating method.

51a ⟨*prc dbl: floating type defs* 50a⟩+≡ (49a) ◁50b 51b▷

```
template <>
const double constants<double>::dxi[] =
{
  3.8060233744356621936e-02,
  1.0838637566236961586e-01,
  1.6221167441072887634e-01,
  1.9134171618254488586e-01,
  1.9134171618254488586e-01,
  1.6221167441072887634e-01,
  1.0838637566236961586e-01,
  3.8060233744356621936e-02
};
```

The data is obviously symmetric, however, it convenient to store all $\Delta\xi_j$ for looping logic. The resulting redundant data is not significant in size and so their storage is not optimized.

Integration weights: The exact integration weights w_{jk}, written to twenty decimal places, are stored in `w` such that each row represents the weights in the inner product for one intra-nodal point as described by Eq. (1.16)

51b ⟨*prc dbl: floating type defs* 50a⟩+≡ (49a) ◁51a

```
template <>
const double constants<double>::w[][n+1] =
{
  {
    1.5446942589330503001e-02, 2.4571482764589171269e-02,
   -2.7292031693117551470e-03, 1.2032717847525820515e-03,
   -7.2548587956445056873e-04, 5.1933327638929719332e-04,
   -4.1852184693699231789e-04, 3.7047163577776345416e-04,
   -1.7805741066949699949e-04
  },
  {
    1.1622746778311606175e-03,  9.0507498067101649271e-02,
    6.1703929899043699124e-02, -1.0149503593034591360e-02,
    5.2541859706611991662e-03, -3.5448091021551241652e-03,
```

```
    2.7783648001647387572e-03, -2.4276059907176542285e-03,
    1.1622746778311606175e-03
  },
  {
    1.3299339017321784808e-02, 6.1002579087781576551e-02,
    1.5851788614537362019e-01, 8.5788009730346129385e-02,
   -1.4811162089693133896e-02, 8.1141101793756148493e-03,
   -5.8380991989182836497e-03, 4.9112819285460210914e-03,
   -2.3256609826782151916e-03
  },
  {
    3.9682539682539682540e-03,  8.1654337316187620786e-02,
    1.2876683494014880164e-01,  1.9930971175315382270e-01,
    9.8412698412698412698e-02, -1.8450782392908931947e-02,
    1.0915704742390880903e-02, -8.5450127081785432803e-03,
    3.9682539682539682540e-03
  },
  {
    1.0262168919186151699e-02, 6.8198042679463056415e-02,
    1.4552063888145796619e-01, 1.7274481918086927590e-01,
    2.1163655891508995929e-01, 9.5070919629898761363e-02,
   -1.8835346462833937648e-02, 1.2106745520227500955e-02,
   -5.3628310808138483005e-03},

  {
    6.7742332586767758905e-03,  7.5536930598726731734e-02,
    1.3690417488237494378e-01,  1.8440373846240001491e-01,
    1.9157121085473562623e-01,  1.9100843295327948211e-01,
    7.7978609783495983416e-02, -1.7398173459092571765e-02,
    6.7742332586767758905e-03
  },

  {
    8.1145653471774335074e-03, 7.2738852972231314052e-02,
    1.4010106152947667486e-01, 1.8033959608385559355e-01,
    1.9755088270496127597e-01, 1.7965565757549230870e-01,
    1.4241174285185143769e-01, 4.8537841843419906236e-02,
   -7.5104346528225664926e-03
  },
  {
    7.9365079365079365079e-03, 7.3109324608009077506e-02,
    1.3968253968253968254e-01, 1.8085892936024489075e-01,
    1.9682539682539682540e-01, 1.8085892936024489075e-01,
    1.3968253968253968254e-01, 7.3109324608009077506e-02,
```

```
    7.9365079365079365079e-03
  }
};
```

3.6.3 Includes & headers

53a ⟨*prc: includes* 53a⟩≡ (47)

```
#include <limits>
#include <cstddef>
```

53b ⟨*prc: header* 53b⟩≡ (47)

```
//=======================================================================
//   Project:  DTIC technical report example integrator implementation
//
// Namespace:  pr_cheby
//
//   Purpose:  specialized constants for Chebyshev integrator of
//             degree n = 8.
//
// $RCSfile: pr-constants.nw,v $
// $Revision: 1.15 $
//
// $State: Exp $
// $Locker:  $
//
// $Source: /home/mitchejw/work/research/tex/dtic/tr/RCS/pr-constants.nw,v $
//
// Comments:  0. Change constants here for the desired type.
//
//   Orig. Author:  Jason Wm. Mitchell
//                  US Air Force Research Laboratory
//                  Air Vehicles Directorate, AFRL/VACA
//                  2210 Eighth Street, Bldg. 146, Rm. 305
//                  Wright-Patterson AFB, OH  45433
//         E-Mail:  jason+dtic_report2002@maiar.org
//     Orig. Date:  02-Feb-2002
//   Last Changed:  $Date: 2002/05/29 15:05:23 $
//
// Copyright (C) 2002 Jason Wm. Mitchell, All Rights Reserved.
// Restrictions:  released under the Perl Artistic license
// Security:      unclassified
//
// This software is OSI Certified Open Source Software.
```

```
// OSI Certified is a certification mark of the Open Source Initiative.
//==========================================================================
```

3.7 Root chunk: pr_cheby

With the supporting code defined, we move on to the implementation of the example integrator.

54a ⟨pr_cheby 54a⟩≡

```
⟨pr: header 65⟩
#ifndef PANOVSKY_RICHARDSON_CHEBYSHEV_8TH_DEGREE
#define PANOVSKY_RICHARDSON_CHEBYSHEV_8TH_DEGREE
⟨pr: includes 64c⟩

namespace pr_cheby
{
  using namespace std;

  ⟨pr: class decl 54b⟩
  ⟨pr: member defs 58a⟩
}
#endif // PANOVSKY_RICHARDSON_CHEBYSHEV_8TH_DEGREE
```

3.7.1 Class declaration

54b ⟨*pr: class decl* 54b⟩≡ (54a)

```
template <typename T, typename EqsT, typename GuessT = default_guesser<T> >
class integrator
{
 public:
  ⟨pr: public typedef decls 56d⟩
  ⟨pr: enumerated status codes 57a⟩
  ⟨pr: constructor decl 57b⟩
  ⟨pr: member-template operator()() decl 57c⟩
 private:
  ⟨pr: private typedef decls 55a⟩
  ⟨pr: private data decls 55b⟩
  ⟨pr: private function decls 56b⟩
};
```

The `integrator` is parameterized by the desired scalar type `T`, as well as by the type of equations and the guesser, `EqsT` and `GuessT`, respectively. The template parameter `GuessT` has as a default argument the previously defined `default_guesser`

object. Supplying a default type to `GuessT` will provide a less cumbersome interface when using the default guesser.

3.7.1.1 Private data

55a ⟨*pr: private* `typedef` *decls* 55a⟩≡ (54b)

```
typedef typename constants<T>::array_type array_type;
```

The `array_type typedef` is forwarded from the traits template `constants` for local use.

55b ⟨*pr: private data decls* 55b⟩≡ (54b) 55c ▷

```
static const size_t n_ = constants<T>::n;
const T*            xi_;
const T*            dxi_;
const array_type    w_;
size_t              iter_max_;
T                   epsilon_;
```

To eliminate the scoping prefix when accessing the specialized traits template `constants`, we declare local shorthands for each identifier but with the established access membership suffix. In the case of the method degree, its value is assigned directly since it is constant and of integral type. Although `iter_max_` is of integral type, its value is not assigned directly, allowing the possibility of assignment at construction. Also, `iter_max_` is not `const`-qualified since it may be useful to alter the value based on convergence performance.

55c ⟨*pr: private data decls* 55b⟩+≡ (54b) ◁ 55b 55d ▷

```
static const size_t o_ = n_ + 1;
```

As shown by Figure 1.1, the total number of nodal points in a given interval is $n + 1$, so that value is assigned here as `o_`.

55d ⟨*pr: private data decls* 55b⟩+≡ (54b) ◁ 55c 55e ▷

```
const size_t m_;
const T      h_;
```

The number of first-order equations `m_` and the fixed stepsize `h_` are declared and `const`-qualified.

55e ⟨*pr: private data decls* 55b⟩+≡ (54b) ◁ 55d 56a ▷

```
valarray<T>             f_arena_;
slice_iterator<T>       f_[ o_ ];
valarray<valarray<T> > y_;
```

The declarations above are responsible for storing the computed derivative and solution values, `f_arena_` and `y_` respectively, over the interval.

Regarding `f_arena_`, recall that Eq. (1.16) requires an inner product operation involving the weights w_{jk} and the derivative values f_k. These weights are stored in the traits template `constants` such that each row represents the coefficients for one first-order equation, y_j. To efficiently use the STL algorithm `inner_product`, the derivative information should also be stored in the same sense, i.e. one row of derivatives should represent the derivative values at each node point for the first-order equation of interest. However, evaluating the system of first-order equations will provide values for all the first-order equations at a given node—the transpose of our desired situation. To handle this, a block of memory large enough to hold all the derivative information is allocated as a `valarray` and identified as `f_arena_`. This memory is handed to the system of first-order equations as a `slice_iterator` whose `slice` correctly spans the arena so that every *row* of `f_arena_` represents the derivative values at the nodal points for the first-order equation of interest. Since there are $n+1$ points, we allocate `o_` iterators.

56a ⟨*pr: private data decls* 55b⟩+≡ (54b) ◁55e

```
EqsT   eqs_;
GuessT guess_;
```

Lastly, it is convenient to contain copies of the first-order equations and the guesser. The overhead associated with retaining copies of these objects is small and only occurs at `integrator` construction.

3.7.1.2 Private functions

56b ⟨*pr: private function decls* 56b⟩≡ (54b) 56c▷

```
void show_static_coefficients_() const;
```

For debugging purposes, a private method is provided to output the coefficients imported from the traits template `constants`.

56c ⟨*pr: private function decls* 56b⟩+≡ (54b) ◁56b

```
integrator();
integrator( const integrator& );
```

The remaining private declarations enforce a ban on default and copy construction for `integrator`.

3.7.1.3 Public `typedefs`

56d ⟨*pr: public* `typedef` *decls* 56d⟩≡ (54b)

```
typedef T      value_type;
typedef EqsT   equation_type;
typedef GuessT guesser_type;
```

The input template parameters are captured in the above public `typedef` declarations.

3.7.1.4 Convergence codes

From Algorithm 1.1, we must have some method to indicate that a particular iteration sequence either converged or failed to converge. This is handled with an enumeration.

57a ⟨*pr: enumerated status codes* 57a⟩≡ (54b)
```
enum status_code_t { CONVERGENCE_FAILED = -1, OK = 0 };
```

Convergence is given the label `OK` and assigned a value of zero. Convergence failure is indicated by a value of `-1` is given the label `CONVERGENCE_FAILED`.

3.7.1.5 Constructor

To construct an instance of `integrator`, we require the fixed stepsize, the system of first-order equations, and a guesser.

57b ⟨*pr: constructor decl* 57b⟩≡ (54b)
```
integrator( const T& stepsize, const EqsT& eqs,
            const GuessT& guess = default_guesser<T>(EqsT::size()) );
```

With this default argument declaration, we see an additional requirement is added to the interface to the first-order equations. The object must provide the `static` member method `size()` to return the number of first-order equations involved.

3.7.1.6 Member-template `operator()()`

The function-like behaviour here is provided by the member-template `operator()()`.

57c ⟨*pr: member-template* `operator()()` *decl* 57c⟩≡ (54b)
```
template <typename V>
int operator()( const T& x, V& yn );
```

As discussed in § 3.4.1.5, the additional type `V` is intended to indicate that `yn` should behave like a mathematical vector, and provide an `operator[]()` random access method.

The argument list expects the current value of the independent variable `x` and the value of the current state `yn`. The type `V` is not `const`-qualified for `yn` since the input state will be over-written with the estimate of the state at $x + h$. The value of `x` is unchanged and thus `const`-qualified, thus `x` must be adjusted by the user for the next step. A return type of `int` is used to return the convergence status of the method invocation. The implicit conversion of the returned enumerated convergence status code of type `status_code_t` is understood and acceptable.

3.7.2 Member definitions

58a ⟨*pr: member defs* 58a⟩≡ (54a)

```
⟨pr: constructor def 58b⟩
⟨pr: debug support def 59d⟩
⟨pr: member-template operator()() def 60c⟩
```

3.7.2.1 Constructor

58b ⟨*pr: constructor def* 58b⟩≡ (58a)

```
template <typename T, typename EqsT, typename GuessT>
inline
integrator<T,EqsT,GuessT>::integrator( const T& stepsize,
                                        const EqsT& eqs,
                                        const GuessT& guess )
⟨pr: c'tor list 58c⟩
{
  ⟨pr: c'tor body 58d⟩
}
```

As previously established in § 3.7.1.5, the `guess` argument is given a default value of `default_guesser<T>` if it is not explicitly provided.

58c ⟨*pr: c'tor list* 58c⟩≡ (58b)

```
      : xi_( constants<T>::xi ),
        dxi_( constants<T>::dxi ),
        w_( constants<T>::w ),
        iter_max_( constants<T>::maximum_iterations ),
        epsilon_( constants<T>::epsilon ),
        m_( EqsT::size() ),
        h_( stepsize ),
        f_arena_( EqsT::size() * o_ ),
        y_( valarray<T>(EqsT::size()), o_ ),
        eqs_( eqs ),
        guess_( guess )
```

The constructor list definition is straightforward. First, each shorthand notation presented in § 3.7.1.1 is initialized with its traits template counterpart. Next, the number of first-order equations and stepsize are stored. Then, the storage for the derivative and solution values at the node points is allocated. Finally, the first-order equations and guesser are copied and stored locally.

58d ⟨*pr: c'tor body* 58d⟩≡ (58b) 59a ▷

```
assert( m_ > 0 );
```

After constructor list completion, we verify that the number of first-order equations is strictly positive.

59a ⟨*pr: c'tor body* 58d⟩+≡ (58b) ◁58d 59b▷

```
for( size_t k = 0; k < o_; k++ )
  f_[k] = slice_iterator<T>(&f_arena_, slice(k,m_,o_));
```

The next construction task is to initialize the array of `slice_iterator`s so that they span `f_arena_` in the desired fashion. In this case, each `slice` has a starting index of `k`, with length `m_` and stride of `m_` and `o_`, respectively.

59b ⟨*pr: c'tor body* 58d⟩+≡ (58b) ◁59a 59c▷

```
for( size_t k = 0; k < y_.size(); k++ )
  assert( y_[k].size() != 0 );
```

In the above, we verify that the length of the elements of `y_` are nonzero.

59c ⟨*pr: c'tor body* 58d⟩+≡ (58b) ◁59b

```
#ifdef PR_CHEBY_SHOW_STATIC_COEFFICIENTS
show_static_coefficients_();
#endif // PR_CHEBY_SHOW_STATIC_COEFFICIENTS
```

Lastly, we provide a preprocessor switch to trigger the output the static coefficients contained in `constants`.

3.7.2.2 Debug support

59d ⟨*pr: debug support def* 59d⟩≡ (58a)

```
template <class T, class EqsT, class GuessT>
inline
void integrator<T,EqsT,GuessT>::show_static_coefficients_() const
{
  ⟨pr: debug: show ξ_j 59e⟩
  ⟨pr: debug: show Δξ_j 60a⟩
  ⟨pr: debug: show w_jk 60b⟩
}
```

Since the above function does not change the state of `integrator`, it is `const`-qualified.

59e ⟨*pr: debug: show ξ_j* 59e⟩≡ (59d)

```
for( size_t j = 0; j < n_ - 1; j++ )
  fprintf(stderr,"xi[%2i]:\t% 23.16e\n", j, xi_[j]);

fprintf(stderr,"\n");
```

The interior intra-nodal points ξ_j are output first. These, as well as the remaining constants, are written to `stderr`.

The function `fprintf` is used here rather than `cerr` for performance reasons. Standard C++ stream output for the compiler and library version selected can be up to a factor of eight times slower than `printf`, or C style, output. This includes the case when output is redirected to a null device. In addition, stream output does not allow for the same formating flexibility as C style output. For these reasons, C style output will be used for all output.

60a ⟨*pr: debug: show* $\Delta\xi_j$ 60a⟩≡ (59d)

```
for( size_t j = 0; j < n_ - 2; j++ )
  fprintf(stderr,"dxi[%2i]:\t% 23.16e\n", j, dxi_[j]);

fprintf(stderr,"\n");
```

Next, we output $\Delta\xi_j$.

60b ⟨*pr: debug: show* w_{jk} 60b⟩≡ (59d)

```
for( size_t j = 0; j < n_; j++ )
  for( size_t l = 0; l < o_; l++ )
    fprintf(stderr,"w[%2i,%2i]:\t% 23.16e\n", j,l,w_[j][l]);

fprintf(stderr,"\n");
```

Lastly, we output the integration weights w_{jk}.

3.7.2.3 Member-template `operator()()`

This member-template definition implements an iteration scheme similar to that described by Algorithm 1.1. The difference, of course, being the use of precomputed and exact integration weights.

60c ⟨*pr: member-template* `operator()()` *def* 60c⟩≡ (58a)

```
template <class T, class EqsT, class GuessT>
template <class V>
inline
int integrator<T,EqsT,GuessT>::operator()( const T& x, V& yn )
{
  ⟨pr:oper: local decls 61a⟩
  ⟨pr:oper: store y0 & f0 info 61b⟩
  ⟨pr:oper: generate guesses for yj if j ≠ 0 61c⟩
  ⟨pr:oper: iteration loop 61d⟩
  ⟨pr:oper: post iteration operations 63a⟩
  ⟨pr:oper: return convergence status 64b⟩
}
```

The argument list above is straightforward, and the interface requirements are discussed in the declaration found in § 3.7.1.6.

61a ⟨*pr:oper: local decls* 61a⟩≡ (60c)
```
status_code_t status = OK;
bool   is_converged = false;
size_t iter_count   = 0;
T   sum;
T   residual;
```

Several locally scoped values are declared and initialized here. The first declaration `status` of type `status_code_t` is used to report the convergence status of the iteration sequence. It is initially assumes convergence and is assigned the enumerated value `OK`. The next declaration `is_converged` is a `bool` that is used as a local loop continuation flag. The `iter_count` declaration counts the number of completed iterations. The last two floating-point declarations `sum` and `residual` are used to hold the newly computed solution and the residual between the new and old solution at an intra-nodal point, respectively.

61b ⟨*pr:oper: store* y_0 *&* f_0 *info* 61b⟩≡ (60c)
```
y_[0] = yn;
eqs_( x, yn, f_[0] );
```

Initially, the input solution `yn` is stored in `y_[0]`, and then the derivatives are computed for `yn` and stored in `f_[0]`.

61c ⟨*pr:oper: generate guesses for* y_j *if* $j \neq 0$ 61c⟩≡ (60c)
```
for( size_t j = 0; j < n_; j++ )
  guess_( x + h_*xi_[j], y_[j], h_*dxi_[j], eqs_, y_[j+1] );
```

To bootstrap the iteration process, solution guesses at the intra-nodal points are generated with `guess_`. The supplied guesser must meet the interface defined in § 3.4.1.5 and § 3.4.2.2. The intra-nodal locations and stepsizes are supplied to the guesser in terms of $h\xi_j$. With respect to the iteration, the solution guess computed with input `y_[j]` is stored in `y_[j+1]`.

61d ⟨*pr:oper: iteration loop* 61d⟩≡ (60c)
```
while( !is_converged  && iter_count++ < iter_max_ )
{
  is_converged = true;

  ⟨pr:oper:iter:a: compute derivatives 62a⟩
  ⟨pr:oper:iter:a: compute solution for each equation and node 62b⟩
}
yn = y_[n_];
```

The iteration loop continues only if the solution has neither converged nor reached the maximum number of iterations specified. Inside the loop, convergence is initially assumed to be `true`. This logic may seem unconventional, but convergence fails if the residual at any nodal point for any equation is above the specified iteration tolerance. Thus, it is more convenient to register a single failure rather than success at every nodal point for all equations.

Once the iteration has completed, the new estimate for $\mathbf{y}(x+h)$ is copied back into `yn` before returning to the calling method.

62a ⟨*pr:oper:iter:a: compute derivatives* 62a⟩≡ (61d)

```
for( size_t j = 1; j < n_; j++ )
  eqs_( x + h_*xi_[j], y_[j], f_[j] );
eqs_( x + h_, y_[n_], f_[n_] );
```

Computing the derivatives is straightforward. But, we must remember to compute f_n since $\xi_n = 1$ is not contained in `xi_`.

62b ⟨*pr:oper:iter:a: compute solution for each equation and node* 62b⟩≡ (61d)

```
for( size_t l = 0; l < m_; l++ )
{
  for( size_t j = 0; j < n_; j++ )
  {
    ⟨pr:oper:iter:b: compute y_j 62c⟩
    ⟨pr:oper:iter:b: check residual and store new value 62d⟩
  }
}
```

Within the main iteration loop, the solution for each equation `l` at each nodal point `j` is considered.

62c ⟨*pr:oper:iter:b: compute* y_j 62c⟩≡ (62b)

```
sum = y_[0][l] + h_ *
      inner_product( &w_[j][0], &w_[j][o_], &f_arena_[l*o_], T(0) );
```

The new solution at the current nodal point as prescribed by Eq. (1.16) is computed and stored in `sum`. The STL algorithm `inner_product` is used to handle the inner product between the exact coefficients w_{jk} and the derivatives f_k for each nodal point j.

62d ⟨*pr:oper:iter:b: check residual and store new value* 62d⟩≡ (62b)

```
residual = fabs( sum - y_[j+1][l] );
if( residual > epsilon_ )
  is_converged = false;

y_[j+1][l] = sum;
```

With the newly computed solution stored in `sum`, it is compared to the previous estimate `y_[j+1][l]` and the residual is computed. If the residual is greater than the iteration tolerance specified by `epsilon_`, then the solution is tagged as not yet converged.

After the convergence check, the newly computed solution `sum` replaces the previous value `y_[j+1][l]`.

63a ⟨*pr:oper: post iteration operations* 63a⟩≡ (60c)

```
bool convergence_failed = (iter_count > iter_max_ && residual > epsilon_);

⟨pr:oper: warn if convergence failed 63b⟩
⟨pr:oper: should grow iteration tolerance? 63c⟩
⟨pr:oper: show convergence statistics 64a⟩
```

After the iteration for the interval has completed, several preprocessor selectable options are provided. Two of the options provide status information regarding the iteration. The third option allows the iteration tolerance to be relaxed in a prescribed manner.

Since two of the provided options require the same boolean test, a single `bool` is defined outside of the preprocessor scoped options. The value of `convergence_failed` is considered to be `true` if the maximum iteration count is exceeded and the residual is greater than the desired tolerance.

63b ⟨*pr:oper: warn if convergence failed* 63b⟩≡ (63a)

```
#ifdef PR_CHEBY_CONVERGENCE_WARN
if( convergence_failed )
{
  status = CONVERGENCE_FAILED;
  fprintf(stderr,
          "...::operator()(...): Warning!  Convergence failure!\n");
  fprintf(stderr," x: % 23.16e,  residual: % 23.16e\n", x, residual);
}
#endif // PR_CHEBY_CONVERGENCE_WARN
```

If selected, the above code block will warn of convergence failure and provide the values of the independent variable and the residual. This information is written to `stderr` to avoid injecting warning information into the user data stream on `stdout`[6]. In addition to the report, the value of `status` is updated to reflect the current iteration state.

63c ⟨*pr:oper: should grow iteration tolerance?* 63c⟩≡ (63a)

```
#if defined PR_CHEBY_GROW_EPSILON
static size_t convergence_watch = 0;
if( convergence_failed )
{
```

[6]See § 3.7.2.2 for the discussion regarding the choice of C style output rather than C++ streams.

```
    if( ++convergence_watch >
        constants<T>::maximum_unconverged_iterations )
    {
      epsilon_ *= constants<T>::epsilon_grow_factor;
      fprintf(stderr,
              "...::operator()(...): Warning! Growing epsilon:%23.16e\n",
              epsilon_);
    }
  }
  #endif // PR_CHEBY_GROW_EPSILON
```

If the above option is selected, it will permit the iteration tolerance to be relaxed for the duration of the integration. A `static` value labeled `convergence_watch` keeps track of the number of times convergence has failed throughout the integration. If it fails to converge a number of times specified by `maximum_unconverged_iterations` in the traits template `constants`, the iteration tolerance `epsilon_` is multiplied by `epsilon_grow_factor`, also specified in the `constants` traits template.

64a ⟨*pr:oper: show convergence statistics* 64a⟩≡ (63a)

```
#ifdef PR_CHEBY_SHOW_STATS
fprintf(stderr,"x: % 23.16e, Passes: %3i,  Residual: % 23.16e\n",
        x, iter_count, residual);
#endif // PR_CHEBY_SHOW_STATS
```

The last preprocessor selectable option reports status information on the recently completed iteration. It provides the values of the independent variable, the number of iterations, and the final residual.

64b ⟨*pr:oper: return convergence status* 64b⟩≡ (60c)

```
return( status );
```

Once the sequence is complete, the status is returned to the calling method.

3.7.3 Includes & header

64c ⟨*pr: includes* 64c⟩≡ (54a)

```
#include <cassert>
#include <cstddef>
#include <cstdio>

#include <algorithm>
#include <numeric>
#include <limits>

#include "pr_constants"
#include "default_guesser"
```

```
#include "slice_iterator"
```

65 ⟨*pr: header* 65⟩≡ (54a)

```
//==============================================================================
//   Project:  DTIC technical report example integrator implementation
//
// Namespace:  pr_cheby
//
//   Purpose:  Perform the Chebyshev integration for degree n = 8.
//
// $RCSfile: pr-cheby.nw,v $
// $Revision: 1.9 $
//
// $State: Exp $
// $Locker:  $
//
// $Source: /home/mitchejw/work/research/tex/dtic/tr/RCS/pr-cheby.nw,v $
//
// Comments:  0. This code should (in general) not be touched.  Nothing here is
//                application specific.  If you want to change the guessing
//                mechanism, look at default_guesser.  Precision changes,
//                look at constants.
//
//            1. This code makes no attempt to automatically adjust step size
//                or method order to match truncation error tolerances.
//
//   Orig. Author:  Jason Wm. Mitchell
//                  US Air Force Research Laboratory
//                  Air Vehicles Directorate, AFRL/VACA
//                  2210 Eighth Street, Bldg. 146, Rm. 305
//                  Wright-Patterson AFB, OH  45433
//         E-Mail:  jason+dtic_report2002@maiar.org
//     Orig. Date:  02-Feb-2002
//   Last Changed:  $Date: 2002/05/29 13:45:18 $
//
// Copyright (C) 2002 Jason Wm. Mitchell, All Rights Reserved.
// Restrictions:  released under the Perl Artistic license
// Security:      unclassified
//
// This software is OSI Certified Open Source Software.
// OSI Certified is a certification mark of the Open Source Initiative.
//===============================================================================
```

3.8 Testing

To test the numerical integrators presented in § 3.4 and § 3.7, we require a system of first-order equations. The system used here represents an harmonic oscillator.

3.8.1 Harmonic oscillator

The interface requirements for the system of first-order equations have already been specified in § 3.4.2.2 and § 3.7.1.5.

66a ⟨`harmonic_oscillator` 66a⟩≡

```
⟨ho: header 67a⟩
#include <cstddef>

struct harmonic_oscillator
{
  ⟨ho: report number of equations 66b⟩
  ⟨ho: member-template operator()() 66c⟩
};
```

66b ⟨*ho: report number of equations* 66b⟩≡ (66a)

```
static size_t size() { return( 2 ); }
```

From § 3.7.1.5, the `integrator` constructor requires a `static` method named `size()` that reports the number of first-order equations as a `size_t`.

66c ⟨*ho: member-template* `operator()()` 66c⟩≡ (66a)

```
template <typename T, typename V1, typename V2> inline
void operator()( const T& x, const V1& y, V2& ydot ) const
{
  ydot[0] =  y[1];
  ydot[1] = -y[0];
}
```

From § 3.4.2.2, invoking the system of first-order equations requires a signature given by the above. The use of a member-template allows the flexibility of accepting differing container types in a type-safe manner. Still, these containers must meet the interface requirements for use in this method. In the above, we only require that `V1` and `V2` provide random access operators, and have elements that are assignable and possess unary minus. Since the state is not changed by invoking this method, it is `const`-qualified.

In terms of operation, the value of the independent variable `x` and state `y` are passed into the method, and the programmed derivatives are computed stored in `ydot` for return to the caller.

Regarding calling efficiency, all the arguments are passed by reference for efficiency since this method will be invoked frequently.

3.8.1.1 Header

67a ⟨*ho: header* 67a⟩≡ (66a)

```
//==========================================================================
//   Project:  DTIC technical report example integrator implementation.
//
// Namespace:  <none>
//
//   Purpose:  Harmonic oscillator equations
//
// $RCSfile: pr-mains.nw,v $
// $Revision: 1.5 $
//
// $State: Exp $
// $Locker:  $
//
// $Source: /home/mitchejw/work/research/tex/dtic/tr/RCS/pr-mains.nw,v $
//
// Comments:
//
//   Orig. Author:  Jason Wm. Mitchell
//                  US Air Force Research Laboratory
//                  Air Vehicles Directorate, AFRL/VACA
//                  2210 Eighth Street, Bldg. 146, Rm. 305
//                  Wright-Patterson AFB, OH  45433  USA
//         E-Mail:  jason+dtic_report2002@maiar.org
//     Orig. Date:  04-May-2002
//   Last Changed:  $Date: 2002/05/29 15:07:12 $
//
// Copyright (C) 2002 Jason Wm. Mitchell, All Rights Reserved.
// Restrictions:  released under Perl Artistic license
// Security:      unclassified
//
// This software is OSI Certified Open Source Software.
// OSI Certified is a certification mark of the Open Source Initiative.
//==========================================================================
```

3.8.2 Fourth-order Runge-Kutta method

Before testing the numerical integrator presented in § 3.7, we must test the fourth-order Runge-Kutta algorithm. Using the set of harmonic oscillator equations defined in § 3.8.1, we construct a small program to validate its operation.

67b ⟨`gmain.cc` 67b⟩≡
  ⟨*gm: header* 70⟩

⟨*gm: includes* 69b⟩
⟨*preprocessor check for* `M_PI` 68a⟩
```
int main()
{
  using namespace std;
```
```
  ⟨gm: typedef decls 68b⟩
  ⟨gm: declaration block 68c⟩
  ⟨gm: integration loop 69a⟩

  return( 0 );
}
```

68a ⟨*preprocessor check for* `M_PI` 68a⟩≡ (67b 71a)
```
#ifndef M_PI
  #error "Try #define _XOPEN_SOURCE 500..."
#endif
```

Since we are using equations for an harmonic oscillator, it is convenient to have access to the frequently provided preprocessor constant `M_PI`. The preprocessor flags that enable the `M_` category of constants vary between compilers and so we provide a minimal check as well as a suggested preprocessor flag to enable them.

68b ⟨*gm:* `typedef` *decls* 68b⟩≡ (67b)
```
typedef harmonic_oscillator eqs_t;
typedef pr_cheby::default_guesser<double> guesser_t;
```

These two declarations provide a shorthand for the descriptive but lengthy types used here.

68c ⟨*gm: declaration block* 68c⟩≡ (67b) 68d ▷
```
const size_t m = eqs_t::size();
const double y0[] = { 0.0, 1.0 };
valarray<double> y(y0,m), y_new(m);
```

The first of the declarations creates a shorthand for the number of first-order equations, i.e. 2. Next, we create a standard `double` array for the initial conditions. The values above correspond to the initial conditions $y(0) = 0$, and $y'(0) = 1$. These initial conditions generate the solutions $y(x) = \sin x$, $y'(x) = \cos x$. Knowing this information allows us to compare the numerical solution to an exact solution for error analysis. Lastly, these initial conditions are used to initialize a `valarray` representing the numerical solution vector. A second `valarray` is constructed to hold the output values from the integration.

68d ⟨*gm: declaration block* 68c⟩+≡ (67b) ◁68c

```
const size_t p = 200;
const double m_2pi = M_PI * 2.0;
const double h = m_2pi / p;
const double xf = m_2pi * 100.0;
double x = 0.0;
double u;
guesser_t rk4(m);
```

Here, we choose the stepsize for the test. To do this, we divide one period of the motion into 200 steps. Since the period is 2π, this gives us a step size of $h = 2\pi/200$. With `h` defined, we elect to run the integration for 100 periods starting from $x_0 = 0$.

To prevent the compiler supplied `sin` and `cos` algorithms from suffering error induced by large input arguments, `u` is provided to store the unwound independent variable `x`.

The last declaration in the block instantiates a `default_guesser` labeled `rk4`.

69a ⟨*gm: integration loop* 69a⟩≡ (67b)

```
for( size_t k = 1; x < xf; k++ )
{
  rk4( x, y, h, eqs_t(), y_new );
  y = y_new;
  x = k*h;
  u = h * (k%p);

  printf("% 23.16e  % 23.16e % 23.16e\n", x, y[0]-sin(x), y[1]-cos(x));
}
```

In the integration loop, the value of $\mathbf{y}(x+h)$ is computed and stored in `y_new`. This state is then copied to `y`, and `x` is incremented to reflect its value at the new state. The value of the unwound argument `u` is computed as $u = h \bmod(k, p)$.

The numerical solution is then compared to the exact solution, and that difference is output. As discussed previously in § 3.7.2.2, C style output is used.

3.8.2.1 Results

Compiling and executing this test produces an error that is $\mathcal{O}(h^4)$ for the duration of the integration for both $\mathbf{y}$ and $\mathbf{y}'$.

3.8.2.2 Includes & header

69b ⟨*gm: includes* 69b⟩≡ (67b)

```
#include <cstddef>
#include <cmath>
#include <cstdio>
```

```
#include <valarray>

#include "default_guesser"
#include "harmonic_oscillator"
```

70 ⟨*gm: header* 70⟩≡ (67b)

```
//==============================================================================
//   Project:  DTIC technical report example integrator implementation.
//
// Namespace:  <none>
//
//   Purpose:  Test default guess generator.
//
// $RCSfile: pr-mains.nw,v $
// $Revision: 1.5 $
//
// $State: Exp $
// $Locker:  $
//
// $Source: /home/mitchejw/work/research/tex/dtic/tr/RCS/pr-mains.nw,v $
//
// Comments:  Just a quick test program to check out RK4 guesser:
//            quick, not elegant.
//
//   Orig. Author:  Jason Wm. Mitchell
//                  US Air Force Research Laboratory
//                  Air Vehicles Directorate, AFRL/VACA
//                  2210 Eighth Street, Bldg. 146, Rm. 305
//                  Wright-Patterson AFB, OH  45433  USA
//         E-Mail:  jason+dtic_report2002@maiar.org
//     Orig. Date:  02-Feb-2002
//   Last Changed:  $Date: 2002/05/29 15:07:12 $
//
// Copyright (C) 2002 Jason Wm. Mitchell, All Rights Reserved.
// Restrictions:  released under Perl Artistic license
// Security:      unclassified
//
// This software is OSI Certified Open Source Software.
// OSI Certified is a certification mark of the Open Source Initiative.
//==============================================================================
```

3.8.3 Chebyshev integrator of degree $n = 8$

Finally, we consider a `main` for testing the integrator of interest: Chebyshev of degree $n = 8$. The system of equations used to verify this integrator will the same harmonic oscillator equations defined in § 3.8.1 used in the previous section.

71a ⟨`cmain.cc` 71a⟩≡

```
⟨cm: header 72c⟩
⟨cm: includes 72b⟩
⟨preprocessor check for M_PI 68a⟩
int main()
{
  using namespace std;

  ⟨cm: typedef decls 71b⟩
  ⟨cm: declaration block 71c⟩
  ⟨cm: integration loop 72a⟩

  return( 0 );
}
```

The basic layout of `cmain.cc` is quite similar to that of `gmain.cc` defined in § 3.8.2.

71b ⟨*cm:* `typedef` *decls* 71b⟩≡ (71a)

```
typedef harmonic_oscillator eqs_t;
typedef pr_cheby::integrator<double,eqs_t> integrator_t;
```

We begin by providing a few convenient `typedef`s, and as before the scalar type used is `double`.

71c ⟨*cm: declaration block* 71c⟩≡ (71a) 71d ▷

```
  const size_t p = 64;
  const double m_2pi = M_PI * 2.0;
  const double h = m_2pi / p;
  const double x0 = 0.0;
  const double xf = m_2pi * 500;
```

In the first portion of the declaration block, we set the interval and the stepsize. In this case, we are only considering 64 points in the period, rather than the 200 chosen for the Runge-Kutta algorithm, resulting in a much larger stepsize. In addition, the integration is performed over 500 periods, compared to the 100 periods for the Runge-Kutta algorithm, starting from $x_0 = 0$.

71d ⟨*cm: declaration block* 71c⟩+≡ (71a) ◁ 71c

```
  double y0[] = {0.0, 1.0};
  valarray<double> y( y0, eqs_t::size() );
  integrator_t cheby8( h, eqs_t() );
```

With the interval and stepsize selected, the initial conditions as discussed in § 3.8.2 are set, and the Chebyshev integrator `cheby8` is instantiated. Note that since the default guessing mechanism is used, the syntax that instantiates `cheby8` is quite simple.

72a ⟨*cm: integration loop* 72a⟩≡ (71a)

```
double u, x = x0;
for( size_t k = 1; x < xf; k++ )
{
  cheby8( x, y );
  x = k*h;
  u = h * (k%p);
  printf("% 23.16e % 23.16e % 23.16e\n", x, sin(u)-y[0], cos(u)-y[1]);
}
```

Before the integration loop definition, we declare the unwound angular argument `u`, and the independent variable `x`. As before, the argument `u` is restricted to `u` $\in [0, 2\pi)$.

Within the integration loop, the current independent variable and state are input into `cheby8()()`. On method return, the new state $\mathbf{y}(x+h)$ is returned in `y`. The independent variable `x` is then incremented accordingly. The current value of `x` is unwound and stored in `u`. This information is used to compare the current numerical solution to the exact solution given in § 3.8.2, with the error output on `stdout`.

3.8.3.1 Results

Compiling and executing this test produces a maximum error over the interval that is on the order of the unit-round for the platform chosen[7].

3.8.3.2 Includes & header

72b ⟨*cm: includes* 72b⟩≡ (71a)

```
#include <cstddef>
#include <cmath>
#include <cstdio>

#include <valarray>

#include "pr_cheby"
#include "harmonic_oscillator"
```

72c ⟨*cm: header* 72c⟩≡ (71a)

```
//=============================================================================
//   Project:  DTIC technical report example integrator implementation
```

[7] See § 3.6.2.

```
//
// Namespace:  <none>
//
//   Purpose:  Driver for Chebyshev integration for degree n = 8.
//
// $RCSfile: pr-mains.nw,v $
// $Revision: 1.5 $
//
// $State: Exp $
// $Locker:  $
//
// $Source: /home/mitchejw/work/research/tex/dtic/tr/RCS/pr-mains.nw,v $
//
// Comments:
//
//   Orig. Author:  Jason Wm. Mitchell
//                  US Air Force Research Laboratory
//                  Air Vehicles Directorate, AFRL/VACA
//                  2210 Eighth Street, Bldg. 146, Rm. 305
//                  Wright-Patterson AFB, OH  45433
//         E-Mail:  jason+dtic_report2002@maiar.org
//     Orig. Date:  02-Feb-2002
//   Last Changed:  $Date: 2002/05/29 15:07:12 $
//
// Copyright (C) 2002 Jason Wm. Mitchell, All Rights Reserved.
// Restrictions:  released under the Perl artistic license
// Security:      unclassified
//
// This software is OSI Certified Open Source Software.
// OSI Certified is a certification mark of the Open Source Initiative.
//===============================================================================
```

Bibliography

[1] K. E. Atkinson. *An introduction to numerical analysis*. John Wiley & Sons Inc., New York, 1989. ISBN 0-471-62489-6.

[2] D. L. Carpenter. *Two families of methods for solving first and second order differential equations using Chebyshev polynomials*. PhD thesis, University of Cincinnati, 1996.

[3] P. Henrici. *Discrete variable methods in ordinary differential equations*. John Wiley & Sons Inc., New York, 1962.

[4] D. E. Knuth. Literate programming. *The Computer Journal*, 27(2):97–111, 1984.

[5] J. Panovsky and D. L. Richardson. A family of implicit Chebyshev methods for the numerical integration of second-order differential equations. *J. Comput. Appl. Math.*, 23(1):35–51, 1988. ISSN 0377-0427.

[6] N. Ramsey. Literate programming simplified. *IEEE Software*, 11(5):97–105, 1994.

[7] D. L. Richardson and J. Panovsky. A family of implicit Chebyshev methods for the numerical integration of first-order differential equations. In *Astrodynamics Specialists 1993*, volume 85 of *Advances in Astronautical Sciences*, pages 2343–2356, San Diego, California, 1994. Univelt.

[8] D. L. Richardson, D. S. Schmidt, and J. W. Mitchell. Improved Chebyshev methods for the numerical integration of first-order differential equations. In *Spaceflight Mechanics 1998*, volume 99 of *Advances in Astronautical Sciences*, pages 1533–1544, San Diego, California, 1998. Univelt.

[9] B. Stroustrup. *The C++ programming language*. Addison-Wesley, Reading, Mass., third (special) edition, 1997. ISBN 0-201-88954-4.

[10] S. Wolfram. *The* MATHEMATICA® *book*. Wolfram Media, Inc., Champaign, IL, 1999. ISBN 1-57955-004-5; 0-521-64314-7.

Appendix A

Intra-nodal points

The following tables contain the intra-nodal point locations, Eq. (1.2) for even degrees $2 \leq n \leq 14$. For each degree, the intra-nodal point locations are given as multiples of a method's fixed stepsize h, i.e. $h_j = h\xi_j$. The values for ξ_0 and ξ_n are not shown in the following since their values are 0 and 1, respectively.

When possible, all numeric quantities in the following tables have been represented as rational functions of nested roots. Unfortunately, this explicit form is not always obtainable in general, so we must consider an implicit symbolic representation of these numbers, something MATHEMATICA® terms *algebraic numbers*[1]. These numbers have the property that applying algebraic operations on them results in a single algebraic number. They are written `Root[g,k]` meaning the k^{th} root of the polynomial equation $g(x) = 0$. For example, `Root[5 − 2 #1^3, 2]` represents the second root of $2x^3 - 5 = 0$, where #1 is the MATHEMATICA® notation for a polynomial variable of interest. This requires a unique ordering of the roots to be meaningful. MATHEMATICA® orders roots by magnitude except that real roots come before complex ones, and complex conjugate pairs are adjacent.

The intra-nodal information presented in the following tables are common for both families of implicit Chebyshev methods.

Table A.1: Intra-nodal points ξ_j, $n = 2$.

j	ξ_j
1	$\frac{1}{2}$

[1]See Wolfram [10, § 3.4.3].

Table A.2: Intra-nodal points ξ_j, $n = 4$.

j	ξ_j	j	ξ_j
1	$\frac{2-\sqrt{2}}{4}$	3	$\frac{2+\sqrt{2}}{4}$
2	$\frac{1}{2}$		

Table A.3: Intra-nodal points ξ_j, $n = 6$.

j	ξ_j	j	ξ_j
1	$\frac{2-\sqrt{3}}{4}$	4	$\frac{3}{4}$
2	$\frac{1}{4}$	5	$\frac{2+\sqrt{3}}{4}$
3	$\frac{1}{2}$		

Table A.4: Intra-nodal points ξ_j, $n = 8$.

j	ξ_j	j	ξ_j
1	$\frac{2-\sqrt{2+\sqrt{2}}}{4}$	5	$\frac{2+\sqrt{2-\sqrt{2}}}{4}$
2	$\frac{2-\sqrt{2}}{4}$	6	$\frac{2+\sqrt{2}}{4}$
3	$\frac{2-\sqrt{2-\sqrt{2}}}{4}$	7	$\frac{2+\sqrt{2+\sqrt{2}}}{4}$
4	$\frac{1}{2}$		

Table A.5: Intra-nodal points ξ_j, $n = 10$.

j	ξ_j	j	ξ_j
1	$\dfrac{4-\sqrt{2\left(5+\sqrt{5}\right)}}{8}$	6	$\dfrac{3+\sqrt{5}}{8}$
2	$\dfrac{3-\sqrt{5}}{8}$	7	$\dfrac{4+\sqrt{10-2\sqrt{5}}}{8}$
3	$\dfrac{4-\sqrt{10-2\sqrt{5}}}{8}$	8	$\dfrac{5+\sqrt{5}}{8}$
4	$\dfrac{5-\sqrt{5}}{8}$	9	$\dfrac{4+\sqrt{2\left(5+\sqrt{5}\right)}}{8}$
5	$\dfrac{1}{2}$		

Table A.6: Intra-nodal points ξ_j, $n = 12$.

j	ξ_j	j	ξ_j
1	$\dfrac{4-\sqrt{2}-\sqrt{6}}{8}$	7	$\dfrac{4-\sqrt{2}+\sqrt{6}}{8}$
2	$\dfrac{2-\sqrt{3}}{4}$	8	$\dfrac{3}{4}$
3	$\dfrac{2-\sqrt{2}}{4}$	9	$\dfrac{2+\sqrt{2}}{4}$
4	$\dfrac{1}{4}$	10	$\dfrac{2+\sqrt{3}}{4}$
5	$\dfrac{4+\sqrt{2}-\sqrt{6}}{8}$	11	$\dfrac{4+\sqrt{2}+\sqrt{6}}{8}$

(*continues*)

Table A.6: Intra-nodal abscissa ξ_j, $n = 12$. (*continued*)

j	ξ_j	j	ξ_j
6	$\frac{1}{2}$		

Table A.7: Intra-nodal points ξ_j, $n = 14$.

j	ξ_j	j	ξ_j
1	$\frac{1-c_6}{2}$	8	$\frac{1+c_1}{2}$
2	$\frac{1-c_5}{2}$	9	$\frac{1+c_2}{2}$
3	$\frac{1-c_4}{2}$	10	$\frac{1+c_3}{2}$
4	$\frac{1-c_3}{2}$	11	$\frac{1+c_4}{2}$
5	$\frac{1-c_2}{2}$	12	$\frac{1+c_5}{2}$
6	$\frac{1-c_1}{2}$	13	$\frac{1+c_6}{2}$
7	$\frac{1}{2}$		

Table A.8: Unique `Root` objects $n = 14$.

ℓ	c_ℓ
1	$\texttt{Root}[1 - 4\,\#1 - 4\,\#1^2 + 8\,\#1^3,\ 2]$
2	$\texttt{Root}[-7 + 56\,\#1^2 - 112\,\#1^4 + 64\,\#1^6,\ 4]$

(*continues*)

Table A.8: Algebraic numbers, $n = 14$. (*continued*)

ℓ	c_ℓ
3	$\texttt{Root}[-1 - 4\,\#1 + 4\,\#1^2 + 8\,\#1^3,\ 3]$
4	$\texttt{Root}[-7 + 56\,\#1^2 - 112\,\#1^4 + 64\,\#1^6,\ 5]$
5	$\texttt{Root}[1 - 4\,\#1 - 4\,\#1^2 + 8\,\#1^3,\ 3]$
6	$\texttt{Root}[-7 + 56\,\#1^2 - 112\,\#1^4 + 64\,\#1^6,\ 6]$

Appendix B

Coefficients for first-order methods

In the following tables, exact coefficients for a family of implicit Chebyshev methods for the numerical integration of first-order differential equations are given. These coefficients for even degrees $2 \leq n \leq 14$ are used to recast Eq. (1.5), such that

$$y\left(x + h\xi_j\right) \; = y(x) + I_j, \tag{B.1}$$

where

$$I_j \; = h\, d_j \sum_{k=0}^{n} w_{jk}\, f(x_k), \tag{B.2}$$

where d_j represents a common denominator for each expression.

The constants c_ℓ appearing in Table B.7 are defined in Appendix A, Table A.8.

Table B.1: Integral approximation I_j for $n = 2$.

d_j, w_{jk}	Expression	w_{jk}	Expression
d_1	$\frac{1}{24}$	$w_{1,1}$	8
$w_{1,0}$	5	$w_{1,2}$	-1
d_2	$\frac{1}{6}$	$w_{2,1}$	4
$w_{2,0}$	1	$w_{2,2}$	1

Table B.2: Integral approximation I_j for $n = 4$.

d_j, w_{jk}	Expression	w_{jk}	Expression
d_1	$\frac{1}{480}$	$w_{1,2}$	$24\left(4 - 3\sqrt{2}\right)$
$w_{1,0}$	$23 + 4\sqrt{2}$	$w_{1,3}$	$64 - 43\sqrt{2}$
$w_{1,1}$	$64 - 13\sqrt{2}$	$w_{1,4}$	$-7 + 4\sqrt{2}$
d_2	$\frac{1}{120}$	$w_{2,2}$	24
$w_{2,0}$	2	$w_{2,3}$	$16 - 15\sqrt{2}$
$w_{2,1}$	$16 + 15\sqrt{2}$	$w_{2,4}$	2
d_3	$\frac{1}{480}$	$w_{3,2}$	$24\left(4 + 3\sqrt{2}\right)$
$w_{3,0}$	$23 - 4\sqrt{2}$	$w_{3,3}$	$64 + 13\sqrt{2}$
$w_{3,1}$	$64 + 43\sqrt{2}$	$w_{3,4}$	$-7 - 4\sqrt{2}$
d_4	$\frac{1}{30}$	$w_{4,2}$	12
$w_{4,0}$	1	$w_{4,3}$	8
$w_{4,1}$	8	$w_{4,4}$	1

Table B.3: Integral approximation I_j for $n = 6$.

d_j, w_{jk}	Expression	w_{jk}	Expression
d_1	$\frac{1}{20160}$	$w_{1,3}$	$2624 - 1488\sqrt{3}$
$w_{1,0}$	$424 + 72\sqrt{3}$	$w_{1,4}$	$2339 - 1368\sqrt{3}$
$w_{1,1}$	$1280 - 235\sqrt{3}$	$w_{1,5}$	$1280 - 725\sqrt{3}$
$w_{1,2}$	$2269 - 1368\sqrt{3}$	$w_{1,6}$	$-136 + 72\sqrt{3}$
d_2	$\frac{1}{20160}$	$w_{2,3}$	-368
$w_{2,0}$	72	$w_{2,4}$	207
$w_{2,1}$	$5\left(296 + 189\sqrt{3}\right)$	$w_{2,5}$	$1480 - 945\sqrt{3}$
$w_{2,2}$	2097	$w_{2,6}$	72
d_3	$\frac{1}{2520}$	$w_{3,3}$	328
$w_{3,0}$	53	$w_{3,4}$	-62
$w_{3,1}$	$10\left(16 + 7\sqrt{3}\right)$	$w_{3,5}$	$160 - 70\sqrt{3}$
$w_{3,2}$	638	$w_{3,6}$	-17
d_4	$\frac{1}{2240}$	$w_{4,3}$	624
$w_{4,0}$	24	$w_{4,4}$	279

(*continues*)

Table B.3: Integral approximation I_j, $n = 6$. (*continued*)

d_j, w_{jk}	Expression	w_{jk}	Expression
$w_{4,1}$	$15\left(8 + 7\sqrt{3}\right)$	$w_{4,5}$	$3\left(40 - 35\sqrt{3}\right)$
$w_{4,2}$	489	$w_{4,6}$	24
d_5	$\frac{1}{20160}$	$w_{5,3}$	$2624 + 1488\sqrt{3}$
$w_{5,0}$	$424 - 72\sqrt{3}$	$w_{5,4}$	$2339 + 1368\sqrt{3}$
$w_{5,1}$	$5\left(256 + 145\sqrt{3}\right)$	$w_{5,5}$	$5\left(256 + 47\sqrt{3}\right)$
$w_{5,2}$	$2269 + 1368\sqrt{3}$	$w_{5,6}$	$-8\left(17 + 9\sqrt{3}\right)$
d_6	$\frac{1}{630}$	$w_{6,3}$	164
$w_{6,0}$	9	$w_{6,4}$	144
$w_{6,1}$	80	$w_{6,5}$	80
$w_{6,2}$	144	$w_{6,6}$	9

Table B.4: Integral approximation I_j for $n = 8$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{40320}$
$w_{1,0}$	$475 + 80\sqrt{2 + \sqrt{2}}$

(*continues*)

Table B.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,1}$	$2560 - 768\sqrt{2} + 126\sqrt{4-2\sqrt{2}} - 210\sqrt{2-\sqrt{2}} - 715\sqrt{2+\sqrt{2}} + 330\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{1,2}$	$1155\sqrt{2} - 16\left(-71 + 42\sqrt{2-\sqrt{2}} + 80\sqrt{2+\sqrt{2}}\right)$
$w_{1,3}$	$768\sqrt{2} - 126\sqrt{4-2\sqrt{2}} + 5\left(512 + 105\sqrt{2-\sqrt{2}} - 206\sqrt{2+\sqrt{2}} - 150\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,4}$	$32\left(124 + 42\sqrt{2-\sqrt{2}} - 85\sqrt{2+\sqrt{2}}\right)$
$w_{1,5}$	$768\sqrt{2} + 294\sqrt{4-2\sqrt{2}} - 5\left(-512 + 105\sqrt{2-\sqrt{2}} + 290\sqrt{2+\sqrt{2}} + 66\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,6}$	$-1155\sqrt{2} - 16\left(-281 + 42\sqrt{2-\sqrt{2}} + 80\sqrt{2+\sqrt{2}}\right)$
$w_{1,7}$	$-768\sqrt{2} - 294\sqrt{4-2\sqrt{2}} + 5\left(512 + 42\sqrt{2-\sqrt{2}} - 353\sqrt{2+\sqrt{2}} + 150\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,8}$	$5\left(-31 + 16\sqrt{2+\sqrt{2}}\right)$
d_2	$\frac{1}{10080}$
$w_{2,0}$	$-20\left(-2+\sqrt{2}\right)$
$w_{2,1}$	$1168 - 512\sqrt{2} + 105\sqrt{2-\sqrt{2}} + 210\sqrt{2+\sqrt{2}}$

(*continues*)

Table B.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,2}$	$704 - 58\sqrt{2}$
$w_{2,3}$	$-128\sqrt{2} + 7\left(16 + 30\sqrt{2-\sqrt{2}} - 15\sqrt{2+\sqrt{2}}\right)$
$w_{2,4}$	$992 - 664\sqrt{2}$
$w_{2,5}$	$-128\sqrt{2} + 7\left(16 - 30\sqrt{2-\sqrt{2}} + 15\sqrt{2+\sqrt{2}}\right)$
$w_{2,6}$	$704 - 478\sqrt{2}$
$w_{2,7}$	$1168 - 512\sqrt{2} - 105\sqrt{2-\sqrt{2}} - 210\sqrt{2+\sqrt{2}}$
$w_{2,8}$	$-20\left(-2+\sqrt{2}\right)$
d_3	$\dfrac{1}{40320}$
$w_{3,0}$	$475 + 80\sqrt{2-\sqrt{2}}$
$w_{3,1}$	$2560 - 768\sqrt{2} + 330\sqrt{4-2\sqrt{2}} - 1450\sqrt{2-\sqrt{2}}$ $+ 525\sqrt{2+\sqrt{2}} + 294\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{3,2}$	$1155\sqrt{2} + 16\left(281 - 80\sqrt{2-\sqrt{2}} + 42\sqrt{2+\sqrt{2}}\right)$
$w_{3,3}$	$2560 + 768\sqrt{2} - 330\sqrt{4-2\sqrt{2}} - 715\sqrt{2-\sqrt{2}}$ $+ 210\sqrt{2+\sqrt{2}} + 126\sqrt{2\left(2+\sqrt{2}\right)}$

(*continues*)

Table B.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,4}$	$-32\left(-124+85\sqrt{2-\sqrt{2}}+42\sqrt{2+\sqrt{2}}\right)$
$w_{3,5}$	$768\sqrt{2}-294\sqrt{2\left(2+\sqrt{2}\right)} - 5\left(-512+150\sqrt{4-2\sqrt{2}}+353\sqrt{2-\sqrt{2}}+42\sqrt{2+\sqrt{2}}\right)$
$w_{3,6}$	$-1155\sqrt{2}+16\left(71-80\sqrt{2-\sqrt{2}}+42\sqrt{2+\sqrt{2}}\right)$
$w_{3,7}$	$2560-768\sqrt{2}+750\sqrt{4-2\sqrt{2}}-1030\sqrt{2-\sqrt{2}} - 525\sqrt{2+\sqrt{2}}-126\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{3,8}$	$5\left(-31+16\sqrt{2-\sqrt{2}}\right)$
d_4	$\frac{1}{2520}$
$w_{4,0}$	10
$w_{4,1}$	$-48\sqrt{2}+5\left(32-21\sqrt{2-\sqrt{2}}+21\sqrt{2+\sqrt{2}}\right)$
$w_{4,2}$	$176+105\sqrt{2}$
$w_{4,3}$	$48\sqrt{2}+5\left(32+21\sqrt{2-\sqrt{2}}+21\sqrt{2+\sqrt{2}}\right)$
$w_{4,4}$	248
$w_{4,5}$	$48\sqrt{2}-5\left(-32+21\sqrt{2-\sqrt{2}}+21\sqrt{2+\sqrt{2}}\right)$

(*continues*)

Table B.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,6}$	$176 - 105\sqrt{2}$
$w_{4,7}$	$-48\sqrt{2} + 5\left(32 + 21\sqrt{2-\sqrt{2}} - 21\sqrt{2+\sqrt{2}}\right)$
$w_{4,8}$	10
d_5	$\frac{1}{40320}$
$w_{5,0}$	$475 - 80\sqrt{2-\sqrt{2}}$
$w_{5,1}$	$2560 - 768\sqrt{2} - 750\sqrt{4-2\sqrt{2}} + 1030\sqrt{2-\sqrt{2}}$ $+ 525\sqrt{2+\sqrt{2}} + 126\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{5,2}$	$1155\sqrt{2} + 16\left(281 + 80\sqrt{2-\sqrt{2}} - 42\sqrt{2+\sqrt{2}}\right)$
$w_{5,3}$	$2560 + 768\sqrt{2} + 750\sqrt{4-2\sqrt{2}} + 1765\sqrt{2-\sqrt{2}}$ $+ 210\sqrt{2+\sqrt{2}} + 294\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{5,4}$	$32\left(124 + 85\sqrt{2-\sqrt{2}} + 42\sqrt{2+\sqrt{2}}\right)$
$w_{5,5}$	$2560 + 768\sqrt{2} + 330\sqrt{4-2\sqrt{2}} + 715\sqrt{2-\sqrt{2}}$ $- 210\sqrt{2+\sqrt{2}} - 126\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{5,6}$	$-1155\sqrt{2} + 16\left(71 + 80\sqrt{2-\sqrt{2}} - 42\sqrt{2+\sqrt{2}}\right)$

(*continues*)

Table B.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{5,7}$	$2560 - 768\sqrt{2} - 330\sqrt{4-2\sqrt{2}} + 1450\sqrt{2-\sqrt{2}} - 525\sqrt{2+\sqrt{2}} - 294\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{5,8}$	$-5\left(31 + 16\sqrt{2-\sqrt{2}}\right)$
d_6	$\frac{1}{10080}$
$w_{6,0}$	$20\left(2+\sqrt{2}\right)$
$w_{6,1}$	$128\sqrt{2} + 7\left(16 + 15\sqrt{2-\sqrt{2}} + 30\sqrt{2+\sqrt{2}}\right)$
$w_{6,2}$	$704 + 478\sqrt{2}$
$w_{6,3}$	$1168 + 512\sqrt{2} + 210\sqrt{2-\sqrt{2}} - 105\sqrt{2+\sqrt{2}}$
$w_{6,4}$	$992 + 664\sqrt{2}$
$w_{6,5}$	$1168 + 512\sqrt{2} - 210\sqrt{2-\sqrt{2}} + 105\sqrt{2+\sqrt{2}}$
$w_{6,6}$	$704 + 58\sqrt{2}$
$w_{6,7}$	$128\sqrt{2} - 7\left(-16 + 15\sqrt{2-\sqrt{2}} + 30\sqrt{2+\sqrt{2}}\right)$
$w_{6,8}$	$20\left(2+\sqrt{2}\right)$
d_7	$\frac{1}{40320}$

(*continues*)

Table B.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{7,0}$	$475 - 80\sqrt{2+\sqrt{2}}$
$w_{7,1}$	$-768\sqrt{2} + 294\sqrt{4-2\sqrt{2}} + 5\left(512 - 42\sqrt{2-\sqrt{2}} + 353\sqrt{2+\sqrt{2}} - 150\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,2}$	$1155\sqrt{2} + 16\left(71 + 42\sqrt{2-\sqrt{2}} + 80\sqrt{2+\sqrt{2}}\right)$
$w_{7,3}$	$2560 + 768\sqrt{2} - 294\sqrt{4-2\sqrt{2}} + 525\sqrt{2-\sqrt{2}} + 1450\sqrt{2+\sqrt{2}} + 330\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{7,4}$	$32\left(124 - 42\sqrt{2-\sqrt{2}} + 85\sqrt{2+\sqrt{2}}\right)$
$w_{7,5}$	$768\sqrt{2} + 126\sqrt{4-2\sqrt{2}} + 5\left(512 - 105\sqrt{2-\sqrt{2}} + 206\sqrt{2+\sqrt{2}} + 150\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,6}$	$-1155\sqrt{2} + 16\left(281 + 42\sqrt{2-\sqrt{2}} + 80\sqrt{2+\sqrt{2}}\right)$
$w_{7,7}$	$2560 - 768\sqrt{2} - 126\sqrt{4-2\sqrt{2}} + 210\sqrt{2-\sqrt{2}} + 715\sqrt{2+\sqrt{2}} - 330\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{7,8}$	$-5\left(31 + 16\sqrt{2+\sqrt{2}}\right)$
d_8	$\dfrac{1}{630}$
$w_{8,0}$	5

(*continues*)

Table B.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,1}$	$80 - 24\sqrt{2}$
$w_{8,2}$	88
$w_{8,3}$	$80 + 24\sqrt{2}$
$w_{8,4}$	124
$w_{8,5}$	$80 + 24\sqrt{2}$
$w_{8,6}$	88
$w_{8,7}$	$80 - 24\sqrt{2}$
$w_{8,8}$	5

Table B.5: Integral approximation I_j for $n = 10$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{8870400}$
$w_{1,0}$	$224\left(298 + 25\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{1,1}$	$386048 - 78848\sqrt{5} + 8085\sqrt{50 - 10\sqrt{5}} - 20955\sqrt{10 - 2\sqrt{5}}$ $- 48391\sqrt{2\left(5+\sqrt{5}\right)} + 14245\sqrt{10\left(5+\sqrt{5}\right)}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,2}$	$548226 - 64174\sqrt{5} - 26400\sqrt{50-10\sqrt{5}} - 142800\sqrt{2\left(5+\sqrt{5}\right)} + 30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,3}$	$386048 + 78848\sqrt{5} - 10395\sqrt{50-10\sqrt{5}} + 26169\sqrt{10-2\sqrt{5}} - 74725\sqrt{2\left(5+\sqrt{5}\right)} - 32725\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,4}$	$163966\sqrt{5} + 26400\sqrt{50-10\sqrt{5}} - 14\left(-21141 + 10200\sqrt{2\left(5+\sqrt{5}\right)} + 2200\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{1,5}$	$192\left(3624 + 220\sqrt{10-2\sqrt{5}} - 1085\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{1,6}$	$779226 - 51326\sqrt{5} + 26400\sqrt{50-10\sqrt{5}} - 142800\sqrt{2\left(5+\sqrt{5}\right)} - 30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,7}$	$386048 + 78848\sqrt{5} + 10395\sqrt{50-10\sqrt{5}} - 47289\sqrt{10-2\sqrt{5}} - 95515\sqrt{2\left(5+\sqrt{5}\right)} - 16555\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,8}$	$526974 - 48466\sqrt{5} - 26400\sqrt{50-10\sqrt{5}} - 142800\sqrt{2\left(5+\sqrt{5}\right)} + 30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,9}$	$-78848\sqrt{5} - 8085\sqrt{50-10\sqrt{5}} - 165\sqrt{10-2\sqrt{5}} + 13\left(29696 - 9373\sqrt{2\left(5+\sqrt{5}\right)} + 2695\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{1,10}$	$224\left(-98 + 25\sqrt{2\left(5+\sqrt{5}\right)}\right)$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
d_2	$\frac{1}{8870400}$
$w_{2,0}$	$-5600\left(-3+\sqrt{5}\right)$
$w_{2,1}$	$388096 - 58496\sqrt{5} + 28875\sqrt{50-10\sqrt{5}} - 28875\sqrt{10-2\sqrt{5}} + 86625\sqrt{2\left(5+\sqrt{5}\right)} - 17325\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{2,2}$	$521450 - 72690\sqrt{5}$
$w_{2,3}$	$80096 - 51104\sqrt{5} - 17325\sqrt{50-10\sqrt{5}} + 86625\sqrt{10-2\sqrt{5}} + 28875\sqrt{2\left(5+\sqrt{5}\right)} - 28875\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{2,4}$	$130\left(415-87\sqrt{5}\right)$
$w_{2,5}$	$-192\left(-3873+1775\sqrt{5}\right)$
$w_{2,6}$	$400450 - 173010\sqrt{5}$
$w_{2,7}$	$80096 - 51104\sqrt{5} + 17325\sqrt{50-10\sqrt{5}} - 86625\sqrt{10-2\sqrt{5}} - 28875\sqrt{2\left(5+\sqrt{5}\right)} + 28875\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{2,8}$	$636950 - 280590\sqrt{5}$
$w_{2,9}$	$388096 - 58496\sqrt{5} - 28875\sqrt{50-10\sqrt{5}} + 28875\sqrt{10-2\sqrt{5}} - 86625\sqrt{2\left(5+\sqrt{5}\right)} + 17325\sqrt{10\left(5+\sqrt{5}\right)}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,10}$	$-5600\left(-3+\sqrt{5}\right)$
d_3	$\frac{1}{8870400}$
$w_{3,0}$	$224\left(298+25\sqrt{10-2\sqrt{5}}\right)$
$w_{3,1}$	$386048-78848\sqrt{5}+16555\sqrt{50-10\sqrt{5}}-95515\sqrt{10-2\sqrt{5}}+47289\sqrt{2\left(5+\sqrt{5}\right)}+10395\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,2}$	$779226+51326\sqrt{5}+30800\sqrt{50-10\sqrt{5}}-142800\sqrt{10-2\sqrt{5}}+26400\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,3}$	$386048+78848\sqrt{5}-14245\sqrt{50-10\sqrt{5}}-48391\sqrt{10-2\sqrt{5}}+20955\sqrt{2\left(5+\sqrt{5}\right)}+8085\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,4}$	$526974+48466\sqrt{5}-30800\sqrt{50-10\sqrt{5}}-142800\sqrt{10-2\sqrt{5}}-26400\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,5}$	$-192\left(-3624+1085\sqrt{10-2\sqrt{5}}+220\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{3,6}$	$548226+64174\sqrt{5}-30800\sqrt{50-10\sqrt{5}}-142800\sqrt{10-2\sqrt{5}}-26400\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,7}$	$386048+78848\sqrt{5}-35035\sqrt{50-10\sqrt{5}}-121849\sqrt{10-2\sqrt{5}}+165\sqrt{2\left(5+\sqrt{5}\right)}-8085\sqrt{10\left(5+\sqrt{5}\right)}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,8}$	$295974 - 163966\sqrt{5} + 30800\sqrt{50-10\sqrt{5}} - 142800\sqrt{10-2\sqrt{5}} + 26400\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,9}$	$386048 - 78848\sqrt{5} + 32725\sqrt{50-10\sqrt{5}} - 74725\sqrt{10-2\sqrt{5}} - 26169\sqrt{2\left(5+\sqrt{5}\right)} - 10395\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,10}$	$224\left(-98+25\sqrt{10-2\sqrt{5}}\right)$
d_4	$\frac{1}{354816}$
$w_{4,0}$	$-224\left(-5+\sqrt{5}\right)$
$w_{4,1}$	$27680 - 8352\sqrt{5} - 1155\sqrt{50-10\sqrt{5}} - 1155\sqrt{10-2\sqrt{5}} + 3465\sqrt{2\left(5+\sqrt{5}\right)} + 693\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{4,2}$	$40850 - 4958\sqrt{5}$
$w_{4,3}$	$3968\sqrt{5} + 3\left(5120 + 231\sqrt{50-10\sqrt{5}} + 1155\sqrt{10-2\sqrt{5}} + 385\sqrt{2\left(5+\sqrt{5}\right)} + 385\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{4,4}$	$22150 + 1598\sqrt{5}$
$w_{4,5}$	$25920 - 13632\sqrt{5}$
$w_{4,6}$	$17530 - 6718\sqrt{5}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,7}$	$3968\sqrt{5} - 3\left(-5120 + 231\sqrt{50 - 10\sqrt{5}} + 1155\sqrt{10 - 2\sqrt{5}} + 385\sqrt{2\left(5 + \sqrt{5}\right)} + 385\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{4,8}$	$26990 - 11426\sqrt{5}$
$w_{4,9}$	$27680 - 8352\sqrt{5} + 1155\sqrt{50 - 10\sqrt{5}} + 1155\sqrt{10 - 2\sqrt{5}} - 3465\sqrt{2\left(5 + \sqrt{5}\right)} - 693\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{4,10}$	$-224\left(-5 + \sqrt{5}\right)$
d_5	$\frac{1}{138600}$
$w_{5,0}$	1043
$w_{5,1}$	$6032 - 1232\sqrt{5} - 1155\sqrt{10 - 2\sqrt{5}} + 1386\sqrt{2\left(5 + \sqrt{5}\right)}$
$w_{5,2}$	$5859 + 3509\sqrt{5}$
$w_{5,3}$	$6032 + 1232\sqrt{5} + 1386\sqrt{10 - 2\sqrt{5}} + 1155\sqrt{2\left(5 + \sqrt{5}\right)}$
$w_{5,4}$	$10941 + 5269\sqrt{5}$
$w_{5,5}$	10872
$w_{5,6}$	$5859 - 3509\sqrt{5}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{5,7}$	$6032 + 1232\sqrt{5} - 1386\sqrt{10 - 2\sqrt{5}} - 1155\sqrt{2\left(5 + \sqrt{5}\right)}$
$w_{5,8}$	$10941 - 5269\sqrt{5}$
$w_{5,9}$	$6032 - 1232\sqrt{5} + 1155\sqrt{10 - 2\sqrt{5}} - 1386\sqrt{2\left(5 + \sqrt{5}\right)}$
$w_{5,10}$	-343
d_6	$\frac{1}{8870400}$
$w_{6,0}$	$5600\left(3 + \sqrt{5}\right)$
$w_{6,1}$	$80096 + 51104\sqrt{5} - 28875\sqrt{50 - 10\sqrt{5}} - 28875\sqrt{10 - 2\sqrt{5}} + 86625\sqrt{2\left(5 + \sqrt{5}\right)} + 17325\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{6,2}$	$400450 + 173010\sqrt{5}$
$w_{6,3}$	$388096 + 58496\sqrt{5} + 17325\sqrt{50 - 10\sqrt{5}} + 86625\sqrt{10 - 2\sqrt{5}} + 28875\sqrt{2\left(5 + \sqrt{5}\right)} + 28875\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{6,4}$	$10\left(63695 + 28059\sqrt{5}\right)$
$w_{6,5}$	$192\left(3873 + 1775\sqrt{5}\right)$
$w_{6,6}$	$521450 + 72690\sqrt{5}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,7}$	$388096 + 58496\sqrt{5} - 17325\sqrt{50 - 10\sqrt{5}} - 86625\sqrt{10 - 2\sqrt{5}} - 28875\sqrt{2\left(5 + \sqrt{5}\right)} - 28875\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{6,8}$	$130\left(415 + 87\sqrt{5}\right)$
$w_{6,9}$	$80096 + 51104\sqrt{5} + 28875\sqrt{50 - 10\sqrt{5}} + 28875\sqrt{10 - 2\sqrt{5}} - 86625\sqrt{2\left(5 + \sqrt{5}\right)} - 17325\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{6,10}$	$5600\left(3 + \sqrt{5}\right)$
d_7	$\dfrac{1}{8870400}$
$w_{7,0}$	$-224\left(-298 + 25\sqrt{10 - 2\sqrt{5}}\right)$
$w_{7,1}$	$386048 - 78848\sqrt{5} - 32725\sqrt{50 - 10\sqrt{5}} + 74725\sqrt{10 - 2\sqrt{5}} + 26169\sqrt{2\left(5 + \sqrt{5}\right)} + 10395\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{7,2}$	$779226 + 51326\sqrt{5} - 30800\sqrt{50 - 10\sqrt{5}} + 142800\sqrt{10 - 2\sqrt{5}} - 26400\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{7,3}$	$386048 + 78848\sqrt{5} + 35035\sqrt{50 - 10\sqrt{5}} + 121849\sqrt{10 - 2\sqrt{5}} - 165\sqrt{2\left(5 + \sqrt{5}\right)} + 8085\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{7,4}$	$526974 + 48466\sqrt{5} + 30800\sqrt{50 - 10\sqrt{5}} + 142800\sqrt{10 - 2\sqrt{5}} + 26400\sqrt{10\left(5 + \sqrt{5}\right)}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{7,5}$	$192\left(3624 + 1085\sqrt{10-2\sqrt{5}} + 220\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{7,6}$	$548226 + 64174\sqrt{5} + 30800\sqrt{50-10\sqrt{5}} + 142800\sqrt{10-2\sqrt{5}} + 26400\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{7,7}$	$386048 + 78848\sqrt{5} + 14245\sqrt{50-10\sqrt{5}} + 48391\sqrt{10-2\sqrt{5}} - 20955\sqrt{2\left(5+\sqrt{5}\right)} - 8085\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{7,8}$	$295974 - 163966\sqrt{5} - 30800\sqrt{50-10\sqrt{5}} + 142800\sqrt{10-2\sqrt{5}} - 26400\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{7,9}$	$386048 - 78848\sqrt{5} - 16555\sqrt{50-10\sqrt{5}} + 95515\sqrt{10-2\sqrt{5}} - 47289\sqrt{2\left(5+\sqrt{5}\right)} - 10395\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{7,10}$	$-224\left(98 + 25\sqrt{10-2\sqrt{5}}\right)$
d_8	$\frac{1}{354816}$
$w_{8,0}$	$224\left(5+\sqrt{5}\right)$
$w_{8,1}$	$-3968\sqrt{5} + 3\left(5120 + 385\sqrt{50-10\sqrt{5}} - 385\sqrt{10-2\sqrt{5}} + 1155\sqrt{2\left(5+\sqrt{5}\right)} - 231\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{8,2}$	$17530 + 6718\sqrt{5}$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,3}$	$27680 + 8352\sqrt{5} - 693\sqrt{50 - 10\sqrt{5}} + 3465\sqrt{10 - 2\sqrt{5}} + 1155\sqrt{2\left(5 + \sqrt{5}\right)} - 1155\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{8,4}$	$26990 + 11426\sqrt{5}$
$w_{8,5}$	$192\left(135 + 71\sqrt{5}\right)$
$w_{8,6}$	$2\left(20425 + 2479\sqrt{5}\right)$
$w_{8,7}$	$27680 + 8352\sqrt{5} + 693\sqrt{50 - 10\sqrt{5}} - 3465\sqrt{10 - 2\sqrt{5}} - 1155\sqrt{2\left(5 + \sqrt{5}\right)} + 1155\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{8,8}$	$22150 - 1598\sqrt{5}$
$w_{8,9}$	$-3968\sqrt{5} + 3\left(5120 - 385\sqrt{50 - 10\sqrt{5}} + 385\sqrt{10 - 2\sqrt{5}} - 1155\sqrt{2\left(5 + \sqrt{5}\right)} + 231\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{8,10}$	$224\left(5 + \sqrt{5}\right)$
d_9	$\dfrac{1}{8870400}$
$w_{9,0}$	$-224\left(-298 + 25\sqrt{2\left(5 + \sqrt{5}\right)}\right)$
$w_{9,1}$	$-78848\sqrt{5} + 8085\sqrt{50 - 10\sqrt{5}} + 165\sqrt{10 - 2\sqrt{5}} + 13\left(29696 + 9373\sqrt{2\left(5 + \sqrt{5}\right)} - 2695\sqrt{10\left(5 + \sqrt{5}\right)}\right)$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{9,2}$	$548226 - 64174\sqrt{5} + 26400\sqrt{50 - 10\sqrt{5}} + 142800\sqrt{2\left(5+\sqrt{5}\right)} - 30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,3}$	$386048 + 78848\sqrt{5} - 10395\sqrt{50 - 10\sqrt{5}} + 47289\sqrt{10 - 2\sqrt{5}} + 95515\sqrt{2\left(5+\sqrt{5}\right)} + 16555\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,4}$	$163966\sqrt{5} - 26400\sqrt{50 - 10\sqrt{5}} + 14\left(21141 + 10200\sqrt{2\left(5+\sqrt{5}\right)} + 2200\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{9,5}$	$-192\left(-3624 + 220\sqrt{10 - 2\sqrt{5}} - 1085\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{9,6}$	$779226 - 51326\sqrt{5} - 26400\sqrt{50 - 10\sqrt{5}} + 142800\sqrt{2\left(5+\sqrt{5}\right)} + 30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,7}$	$386048 + 78848\sqrt{5} + 10395\sqrt{50 - 10\sqrt{5}} - 26169\sqrt{10 - 2\sqrt{5}} + 74725\sqrt{2\left(5+\sqrt{5}\right)} + 32725\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,8}$	$526974 - 48466\sqrt{5} + 26400\sqrt{50 - 10\sqrt{5}} + 142800\sqrt{2\left(5+\sqrt{5}\right)} - 30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,9}$	$386048 - 78848\sqrt{5} - 8085\sqrt{50 - 10\sqrt{5}} + 20955\sqrt{10 - 2\sqrt{5}} + 48391\sqrt{2\left(5+\sqrt{5}\right)} - 14245\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,10}$	$-224\left(98 + 25\sqrt{2\left(5+\sqrt{5}\right)}\right)$

(*continues*)

Table B.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
d_{10}	$\frac{1}{34650}$
$w_{10,0}$	175
$w_{10,1}$	$3016 - 616\sqrt{5}$
$w_{10,2}$	$4200 - 440\sqrt{5}$
$w_{10,3}$	$3016 + 616\sqrt{5}$
$w_{10,4}$	$40\left(105 + 11\sqrt{5}\right)$
$w_{10,5}$	5436
$w_{10,6}$	$40\left(105 + 11\sqrt{5}\right)$
$w_{10,7}$	$3016 + 616\sqrt{5}$
$w_{10,8}$	$4200 - 440\sqrt{5}$
$w_{10,9}$	$3016 - 616\sqrt{5}$
$w_{10,10}$	175

Table B.6: Integral approximation I_j for $n = 12$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{69189120}$
$w_{1,0}$	$1680\left(215+18\sqrt{2}+18\sqrt{6}\right)$
$w_{1,1}$	$-101677\sqrt{2}-97361\sqrt{6}-2048\left(-1346+455\sqrt{3}\right)$
$w_{1,2}$	$-22\left(-162392+38624\sqrt{2}+33033\sqrt{3}+20736\sqrt{6}\right)$
$w_{1,3}$	$997421\sqrt{2}-1040303\sqrt{6}-2048\left(-1346+455\sqrt{3}\right)$
$w_{1,4}$	$-22\left(-44456+38624\sqrt{2}-33033\sqrt{3}+20736\sqrt{6}\right)$
$w_{1,5}$	$-16\left(-199744+4261\sqrt{2}+78192\sqrt{6}\right)$
$w_{1,6}$	$-6\left(-1027709+248256\sqrt{2}+216216\sqrt{3}+124704\sqrt{6}\right)$
$w_{1,7}$	$-1833043\sqrt{2}-720529\sqrt{6}+2048\left(1346+455\sqrt{3}\right)$
$w_{1,8}$	$64\left(70808+4117\sqrt{2}-31347\sqrt{6}\right)$
$w_{1,9}$	$-1202413\sqrt{2}-1086895\sqrt{6}+2048\left(1346+455\sqrt{3}\right)$
$w_{1,10}$	$6\left(280963-248256\sqrt{2}+216216\sqrt{3}-124704\sqrt{6}\right)$
$w_{1,11}$	$-16\left(-199744+130387\sqrt{2}+6120\sqrt{6}\right)$
$w_{1,12}$	$1680\left(-71+18\sqrt{2}+18\sqrt{6}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_2	$\frac{1}{69189120}$
$w_{2,0}$	$-60480\left(-2+\sqrt{3}\right)$
$w_{2,1}$	$5115328 + 519519\sqrt{2} - 2136256\sqrt{3} + 291291\sqrt{6}$
$w_{2,2}$	$2275328 - 1293534\sqrt{3}$
$w_{2,3}$	$5115328 - 519519\sqrt{2} - 2136256\sqrt{3} - 291291\sqrt{6}$
$w_{2,4}$	$2275328 - 176418\sqrt{3}$
$w_{2,5}$	$-32\left(-99872 + 3003\sqrt{2} + 60804\sqrt{3}\right)$
$w_{2,6}$	$54\left(73705 - 40928\sqrt{3}\right)$
$w_{2,7}$	$397888 - 519519\sqrt{2} - 272576\sqrt{3} + 291291\sqrt{6}$
$w_{2,8}$	$128\left(35404 - 20133\sqrt{3}\right)$
$w_{2,9}$	$397888 + 519519\sqrt{2} - 272576\sqrt{3} - 291291\sqrt{6}$
$w_{2,10}$	$54\left(71703 - 40928\sqrt{3}\right)$
$w_{2,11}$	$32\left(99872 + 3003\sqrt{2} - 60804\sqrt{3}\right)$
$w_{2,12}$	$-60480\left(-2+\sqrt{3}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_3	$\dfrac{1}{4324320}$
$w_{3,0}$	$105\left(215 + 36\sqrt{2}\right)$
$w_{3,1}$	$172288 - 78878\sqrt{2} - 58240\sqrt{3} + 62972\sqrt{6}$
$w_{3,2}$	$176\left(808 + 79\sqrt{2} - 546\sqrt{3}\right)$
$w_{3,3}$	$172288 - 114914\sqrt{2} - 58240\sqrt{3} + 38948\sqrt{6}$
$w_{3,4}$	$176\left(808 + 79\sqrt{2} + 546\sqrt{3}\right)$
$w_{3,5}$	$199744 - 11525\sqrt{2}$
$w_{3,6}$	$233364 - 186192\sqrt{2}$
$w_{3,7}$	$172288 - 114914\sqrt{2} + 58240\sqrt{3} - 38948\sqrt{6}$
$w_{3,8}$	$283232 - 207304\sqrt{2}$
$w_{3,9}$	$172288 - 78878\sqrt{2} + 58240\sqrt{3} - 62972\sqrt{6}$
$w_{3,10}$	$257388 - 186192\sqrt{2}$
$w_{3,11}$	$199744 - 137651\sqrt{2}$
$w_{3,12}$	$105\left(-71 + 36\sqrt{2}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_4	$\frac{1}{69189120}$
$w_{4,0}$	60480
$w_{4,1}$	$417344 + 567567\sqrt{2} + 495040\sqrt{3} + 243243\sqrt{6}$
$w_{4,2}$	$2098624 - 1135134\sqrt{3}$
$w_{4,3}$	$417344 - 567567\sqrt{2} + 495040\sqrt{3} - 243243\sqrt{6}$
$w_{4,4}$	$22\left(95392 + 51597\sqrt{3}\right)$
$w_{4,5}$	$32\left(109996 + 81081\sqrt{2}\right)$
$w_{4,6}$	3746790
$w_{4,7}$	$417344 - 567567\sqrt{2} - 495040\sqrt{3} + 243243\sqrt{6}$
$w_{4,8}$	343936
$w_{4,9}$	$417344 + 567567\sqrt{2} - 495040\sqrt{3} - 243243\sqrt{6}$
$w_{4,10}$	179226
$w_{4,11}$	$32\left(109996 - 81081\sqrt{2}\right)$
$w_{4,12}$	60480

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_5	$\frac{1}{69189120}$
$w_{5,0}$	$-1680\left(-215+18\sqrt{2}-18\sqrt{6}\right)$
$w_{5,1}$	$1833043\sqrt{2}-720529\sqrt{6}-2048\left(-1346+455\sqrt{3}\right)$
$w_{5,2}$	$22\left(44456+38624\sqrt{2}-33033\sqrt{3}-20736\sqrt{6}\right)$
$w_{5,3}$	$1202413\sqrt{2}-1086895\sqrt{6}-2048\left(-1346+455\sqrt{3}\right)$
$w_{5,4}$	$22\left(162392+38624\sqrt{2}+33033\sqrt{3}-20736\sqrt{6}\right)$
$w_{5,5}$	$16\left(199744+130387\sqrt{2}-6120\sqrt{6}\right)$
$w_{5,6}$	$6\left(1027709+248256\sqrt{2}+216216\sqrt{3}-124704\sqrt{6}\right)$
$w_{5,7}$	$101677\sqrt{2}-97361\sqrt{6}+2048\left(1346+455\sqrt{3}\right)$
$w_{5,8}$	$-64\left(-70808+4117\sqrt{2}+31347\sqrt{6}\right)$
$w_{5,9}$	$-997421\sqrt{2}-1040303\sqrt{6}+2048\left(1346+455\sqrt{3}\right)$
$w_{5,10}$	$6\left(280963+248256\sqrt{2}-216216\sqrt{3}-124704\sqrt{6}\right)$
$w_{5,11}$	$16\left(199744+4261\sqrt{2}-78192\sqrt{6}\right)$
$w_{5,12}$	$-1680\left(71+18\sqrt{2}-18\sqrt{6}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_6	$\frac{1}{1081080}$
$w_{6,0}$	1890
$w_{6,1}$	$43072 - 21021\sqrt{2} - 14560\sqrt{3} + 21021\sqrt{6}$
$w_{6,2}$	$35552 - 18018\sqrt{3}$
$w_{6,3}$	$43072 + 21021\sqrt{2} - 14560\sqrt{3} - 21021\sqrt{6}$
$w_{6,4}$	$22\left(1616 + 819\sqrt{3}\right)$
$w_{6,5}$	$49936 + 39039\sqrt{2}$
$w_{6,6}$	115398
$w_{6,7}$	$43072 + 21021\sqrt{2} + 14560\sqrt{3} + 21021\sqrt{6}$
$w_{6,8}$	70808
$w_{6,9}$	$43072 - 21021\sqrt{2} + 14560\sqrt{3} - 21021\sqrt{6}$
$w_{6,10}$	7290
$w_{6,11}$	$49936 - 39039\sqrt{2}$
$w_{6,12}$	1890

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_7	$\frac{1}{69189120}$
$w_{7,0}$	$1680\left(215 + 18\sqrt{2} - 18\sqrt{6}\right)$
$w_{7,1}$	$-1202413\sqrt{2} + 1086895\sqrt{6} - 2048\left(-1346 + 455\sqrt{3}\right)$
$w_{7,2}$	$-22\left(-44456 + 38624\sqrt{2} + 33033\sqrt{3} - 20736\sqrt{6}\right)$
$w_{7,3}$	$-1833043\sqrt{2} + 720529\sqrt{6} - 2048\left(-1346 + 455\sqrt{3}\right)$
$w_{7,4}$	$22\left(162392 - 38624\sqrt{2} + 33033\sqrt{3} + 20736\sqrt{6}\right)$
$w_{7,5}$	$16\left(199744 - 4261\sqrt{2} + 78192\sqrt{6}\right)$
$w_{7,6}$	$6\left(1027709 - 248256\sqrt{2} + 216216\sqrt{3} + 124704\sqrt{6}\right)$
$w_{7,7}$	$997421\sqrt{2} + 1040303\sqrt{6} + 2048\left(1346 + 455\sqrt{3}\right)$
$w_{7,8}$	$64\left(70808 + 4117\sqrt{2} + 31347\sqrt{6}\right)$
$w_{7,9}$	$-101677\sqrt{2} + 97361\sqrt{6} + 2048\left(1346 + 455\sqrt{3}\right)$
$w_{7,10}$	$6\left(280963 - 248256\sqrt{2} - 216216\sqrt{3} + 124704\sqrt{6}\right)$
$w_{7,11}$	$16\left(199744 - 130387\sqrt{2} + 6120\sqrt{6}\right)$
$w_{7,12}$	$1680\left(-71 + 18\sqrt{2} - 18\sqrt{6}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_8	$\frac{1}{2562560}$
$w_{8,0}$	6720
$w_{8,1}$	$3\left(62912 + 7007\sqrt{2} - 29120\sqrt{3} + 3003\sqrt{6}\right)$
$w_{8,2}$	$3\left(30272 - 14014\sqrt{3}\right)$
$w_{8,3}$	$-3\left(-62912 + 7007\sqrt{2} + 29120\sqrt{3} + 3003\sqrt{6}\right)$
$w_{8,4}$	$66\left(1376 + 637\sqrt{3}\right)$
$w_{8,5}$	$96\left(1108 + 1001\sqrt{2}\right)$
$w_{8,6}$	284178
$w_{8,7}$	$3\left(62912 - 7007\sqrt{2} + 29120\sqrt{3} + 3003\sqrt{6}\right)$
$w_{8,8}$	322944
$w_{8,9}$	$3\left(62912 + 7007\sqrt{2} + 29120\sqrt{3} - 3003\sqrt{6}\right)$
$w_{8,10}$	152046
$w_{8,11}$	$106368 - 96096\sqrt{2}$
$w_{8,12}$	6720

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_9	$\frac{1}{4324320}$
$w_{9,0}$	$22575 - 3780\sqrt{2}$
$w_{9,1}$	$2\left(86144 + 57457\sqrt{2} - 29120\sqrt{3} - 19474\sqrt{6}\right)$
$w_{9,2}$	$-176\left(-808 + 79\sqrt{2} + 546\sqrt{3}\right)$
$w_{9,3}$	$2\left(86144 + 39439\sqrt{2} - 29120\sqrt{3} - 31486\sqrt{6}\right)$
$w_{9,4}$	$176\left(808 - 79\sqrt{2} + 546\sqrt{3}\right)$
$w_{9,5}$	$199744 + 137651\sqrt{2}$
$w_{9,6}$	$233364 + 186192\sqrt{2}$
$w_{9,7}$	$2\left(86144 + 39439\sqrt{2} + 29120\sqrt{3} + 31486\sqrt{6}\right)$
$w_{9,8}$	$283232 + 207304\sqrt{2}$
$w_{9,9}$	$2\left(86144 + 57457\sqrt{2} + 29120\sqrt{3} + 19474\sqrt{6}\right)$
$w_{9,10}$	$257388 + 186192\sqrt{2}$
$w_{9,11}$	$199744 + 11525\sqrt{2}$
$w_{9,12}$	$-105\left(71 + 36\sqrt{2}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_{10}	$\frac{1}{69189120}$
$w_{10,0}$	$60480\left(2+\sqrt{3}\right)$
$w_{10,1}$	$397888+519519\sqrt{2}+272576\sqrt{3}+291291\sqrt{6}$
$w_{10,2}$	$22\left(103424+8019\sqrt{3}\right)$
$w_{10,3}$	$397888-519519\sqrt{2}+272576\sqrt{3}-291291\sqrt{6}$
$w_{10,4}$	$22\left(103424+58797\sqrt{3}\right)$
$w_{10,5}$	$32\left(99872-3003\sqrt{2}+60804\sqrt{3}\right)$
$w_{10,6}$	$54\left(73705+40928\sqrt{3}\right)$
$w_{10,7}$	$5115328-519519\sqrt{2}+2136256\sqrt{3}+291291\sqrt{6}$
$w_{10,8}$	$128\left(35404+20133\sqrt{3}\right)$
$w_{10,9}$	$5115328+519519\sqrt{2}+2136256\sqrt{3}-291291\sqrt{6}$
$w_{10,10}$	$54\left(71703+40928\sqrt{3}\right)$
$w_{10,11}$	$32\left(99872+3003\sqrt{2}+60804\sqrt{3}\right)$
$w_{10,12}$	$60480\left(2+\sqrt{3}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_{11}	$\frac{1}{69189120}$
$w_{11,0}$	$-1680\left(-215+18\sqrt{2}+18\sqrt{6}\right)$
$w_{11,1}$	$-997421\sqrt{2}+1040303\sqrt{6}-2048\left(-1346+455\sqrt{3}\right)$
$w_{11,2}$	$22\left(162392+38624\sqrt{2}-33033\sqrt{3}+20736\sqrt{6}\right)$
$w_{11,3}$	$101677\sqrt{2}+97361\sqrt{6}-2048\left(-1346+455\sqrt{3}\right)$
$w_{11,4}$	$22\left(44456+38624\sqrt{2}+33033\sqrt{3}+20736\sqrt{6}\right)$
$w_{11,5}$	$16\left(199744+130387\sqrt{2}+6120\sqrt{6}\right)$
$w_{11,6}$	$6\left(1027709+248256\sqrt{2}-216216\sqrt{3}+124704\sqrt{6}\right)$
$w_{11,7}$	$1202413\sqrt{2}+1086895\sqrt{6}+2048\left(1346+455\sqrt{3}\right)$
$w_{11,8}$	$64\left(70808-4117\sqrt{2}+31347\sqrt{6}\right)$
$w_{11,9}$	$1833043\sqrt{2}+720529\sqrt{6}+2048\left(1346+455\sqrt{3}\right)$
$w_{11,10}$	$6\left(280963+248256\sqrt{2}+216216\sqrt{3}+124704\sqrt{6}\right)$
$w_{11,11}$	$16\left(199744+4261\sqrt{2}+78192\sqrt{6}\right)$
$w_{11,12}$	$-1680\left(71+18\sqrt{2}+18\sqrt{6}\right)$

(*continues*)

Table B.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
d_{12}	$\frac{1}{270270}$
$w_{12,0}$	945
$w_{12,1}$	$21536 - 7280\sqrt{3}$
$w_{12,2}$	17776
$w_{12,3}$	$21536 - 7280\sqrt{3}$
$w_{12,4}$	17776
$w_{12,5}$	24968
$w_{12,6}$	30672
$w_{12,7}$	$16\left(1346 + 455\sqrt{3}\right)$
$w_{12,8}$	35404
$w_{12,9}$	$16\left(1346 + 455\sqrt{3}\right)$
$w_{12,10}$	30672
$w_{12,11}$	24968
$w_{12,12}$	945

Table B.7: Integral approximation I_j for $n = 14$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{10090080}$
$w_{1,0}$	$132\,(293 + 98\,c_6)$
$w_{1,1}$	$143\,c_1\,(-64 + 119\,c_2) + 7\,(-28\,(-1848 + 208\,c_3 + 429\,c_4 + 1200\,c_5) + 13\,(11\,c_2\,(-21 + 22\,c_3) + 6\,c_4\,(61\,c_3 + 115\,c_5))) + 66\,(-3027 + 2765\,c_5)\,c_6$
$w_{1,2}$	$-105105\,{c_1}^2 + 286\,c_3\,(-817 + 392\,c_2 + 294\,c_5) + 294\,c_1\,(344 + 429\,c_3 + 520\,c_4 - 1320\,c_6) + 49\,(26\,c_5\,(-32 + 88\,c_2 - 120\,c_4 + 165\,c_5) - 231\,(-33 + 32\,c_6))$
$w_{1,3}$	$-2002\,c_2\,(42 + 43\,c_3 - 91\,c_5) + 1568\,(231 + 26\,c_5) + 390\,c_4\,(-407 + 161\,c_5) - 105\,c_1\,(2240 + 832\,c_4 - 3025\,c_6) - 379533\,c_6 + 286\,c_3\,(32 + 189\,c_6)$
$w_{1,4}$	$91\,c_1\,(2819 + 1232\,c_2 + 231\,c_3 - 1680\,c_4 - 2310\,c_5) + 21\,c_3\,(-13539 + 6006\,c_3 - 7280\,c_4 + 18480\,c_6) - 11\,(26\,(593 + 392\,c_2 - 735\,c_5)\,c_5 + 21\,(-1487 + 1568\,c_6))$
$w_{1,5}$	$143\,c_1\,(-64 + 329\,c_2) - 1001\,c_2\,(-21 + 86\,c_3) + 98\,(3696 - 416\,c_3 + 858\,c_4 + 1443\,c_3\,c_4) - 210\,(1120 + 1131\,c_4)\,c_5 + 66\,(-7837 + 7315\,c_5)\,c_6$
$w_{1,6}$	$637\,c_1\,(-229 + 176\,c_2 - 33\,c_3 - 240\,c_4 + 330\,c_5) - 147\,c_3\,(1403 + 858\,c_3 + 1040\,c_4 - 2640\,c_6) - 11\,(26\,c_5\,(-817 + 392\,c_2 + 735\,c_5) + 1029\,(-33 + 32\,c_6))$
$w_{1,7}$	$9152\,c_3 + 390\,c_4\,(407 - 609\,c_5) + 2002\,c_2\,(42 + 11\,c_3 - 59\,c_5) + 1568\,(231 + 26\,c_5) - 105\,c_1\,(2240 + 832\,c_4 - 3311\,c_6) - 693\,(487 + 78\,c_3)\,c_6$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,8}$	$169598\, c_3 + 7\, (21632\, c_5 + 33\, (1487 - 1568\, c_6)) + 7\, \left(15015\, {c_1}^2 + 182\, (88\, c_2\, (c_3 + c_5) - 3\, c_5\, (22\, c_3 + 40\, c_4 + 55\, c_5)) - 6\, c_1\, (-9272 + 3003\, c_3 - 3640\, c_4 + 9240\, c_6)\right)$
$w_{1,9}$	$362208 + 235200\, c_3 + 91\, c_1\, (448 + 352\, c_2 - 1125\, c_4) - 21021\, c_4 + 141414\, c_3\, c_4 - 9152\, c_5 - 286\, c_2\, (-555 + 413\, c_5) - 462\, (594 + 603\, c_3 + 325\, c_5)\, c_6$
$w_{1,10}$	$-143\, c_1\, (-899 + 784\, c_2 + 2205\, c_5) + 49\, \left(2574\, {c_3}^2 - 52\, c_3\, (-16 + 44\, c_2 - 60\, c_4 + 33\, c_5) + 3\, (2541 + 1403\, c_5 + 1040\, c_4\, c_5 - 176\, (14 + 15\, c_5)\, c_6)\right)$
$w_{1,11}$	$-16\, (-35362 + 4004\, c_2 - 10920\, c_4 + 43197\, c_6)$
$w_{1,12}$	$-143\, c_1\, (451 + 784\, c_2 - 2205\, c_5) - 7\, \left(18018\, {c_3}^2 + 52\, c_3\, (416 + 308\, c_2 - 420\, c_4 - 231\, c_5) - 3\, (16357 + (13539 + 7280\, c_4)\, c_5) + 3696\, (14 + 15\, c_5)\, c_6\right)$
$w_{1,13}$	$362208 + 235200\, c_3 + 91\, c_1\, (448 + 352\, c_2 - 795\, c_4) + 21021\, c_4 + 33306\, c_3\, c_4 - 9152\, c_5 + 286\, c_2\, (-555 + 637\, c_5) - 462\, (958 + 837\, c_3 - 325\, c_5)\, c_6$
$w_{1,14}$	$132\, (-97 + 98\, c_6)$
d_2	$\dfrac{1}{10090080}$
$w_{2,0}$	$-12936\, (-1 + c_5)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,1}$	$-26\, c_1\, (-3608 + 4851\, c_2 - 1568\, c_3 - 4851\, c_4 + 5880\, c_5) + 49\, (429\, c_2\, (-1 + 5\, c_5) + 2\, (3696 - 858\, c_4 + 24\, c_5\, (-254 + 165\, c_5) + 13\, c_3\, (-32 + 88\, c_3 + 165\, c_4 - 165\, c_6) + 2145\, c_6))$
$w_{2,2}$	$7\, \left(53361 + 4758\, {c_1}^2 + c_1\, (29463 - 50133\, c_5) - 54895\, c_5 + 286\, c_3\, (-91 + 59\, c_3 + 59\, c_5)\right)$
$w_{2,3}$	$-152880\, {c_1}^2 + 42042\, c_2\, (-2 + 5\, c_3) + 176\, (2058 + 13\, c_3\, (-41 + 49\, c_3)) + 210210\, (-1 + c_3)\, c_4 - 784\, (410 + 143\, c_3)\, c_5 + 294\, c_1\, (429\, c_2 + 800\, (-1 + c_5) - 429\, c_6) + 21021\, (-1 + 5\, c_5)\, c_6$
$w_{2,4}$	$-7\, (-49071 + 40617\, c_3 + 26\, c_1\, (-832 + 777\, c_1 + 1778\, c_3) + 70235\, c_5 - 75691\, c_3\, c_5)$
$w_{2,5}$	$26\, c_1\, (3608 + 4851\, c_2 + 1568\, c_3 - 4851\, c_4 - 5880\, c_5) + 49\, (-429\, c_2\, (-1 + 5\, c_5) + 2\, (3\, (1232 + 286\, c_4 + 8\, c_5\, (-254 + 165\, c_5) - 715\, c_6) + 13\, c_3\, (-32 + 88\, c_3 - 165\, c_4 + 165\, c_6)))$
$w_{2,6}$	$-7\, (-53361 + 26\, c_1\, (224 + 183\, c_1 - 466\, c_3) + 29463\, c_3 + 23045\, c_5 - 28501\, c_3\, c_5)$
$w_{2,7}$	$-152880\, {c_1}^2 - 93808\, c_3 - 294\, c_1\, (429\, c_2 - 800\, (-1 + c_5) - 429\, c_6) + 49\, (7392 + 4290\, c_4 - 6560\, c_5 + 429\, c_6) - 7007\, \left(-16\, {c_3}^2 + 6\, c_2\, (-2 + 5\, c_3) + 30\, c_3\, c_4 + 16\, c_3\, c_5 + 15\, c_5\, c_6\right)$
$w_{2,8}$	$7\, \left(49071 + 20202\, {c_1}^2 + c_1\, (40617 - 69867\, c_5) - 31729\, c_5 - 286\, c_3\, (-59 + 91\, c_3 + 91\, c_5)\right)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,9}$	$-152880\, c_1{}^2 + 235200\, c_3 + 1274\, c_1\ (32 + 32\, c_3 - 99\, c_4 - 99\, c_6)$ $- 539\ (390\, c_2\ (-1 + c_3) + 928\, c_3\, c_5 - 195\, c_4\, c_5 + 390\, c_3\, c_6)$ $- 11\ (1911\, c_4 + 4\ (6100\, c_5 - 147\ (56 + 13\, c_6)))$
$w_{2,10}$	$-7\ (26\, c_3\ (-224 + 649\, c_3) + 26\, c_1\ (-1001 + 466\, c_3 - 183\, c_5)$ $+ 3\ (-17787 + c_5\ (6536 + 15125\, c_5)))$
$w_{2,11}$	$16\ (28927 + 19110\, c_1 + 14014\, c_3 - 46893\, c_5)$
$w_{2,12}$	$7\ (338\, c_3\ (-64 + 77\, c_3) + 26\, c_1\ (-649 + 1778\, c_3 + 777\, c_5)$ $- 3\ (-16357 + c_5\ (4248 + 16555\, c_5)))$
$w_{2,13}$	$-152880\, c_1{}^2 + 235200\, c_3 + 11\ (32928 + 1911\, c_4 - 24400\, c_5 - 7644\, c_6)$ $+ 1274\, c_1\ (32 + 32\, c_3 + 99\, c_4 + 99\, c_6)$ $+ 539\ (390\, c_2\ (-1 + c_3) - 928\, c_3\, c_5 - 195\, c_4\, c_5 + 390\, c_3\, c_6)$
$w_{2,14}$	$-12936\ (-1 + c_5)$
d_3	$\dfrac{1}{10090080}$
$w_{3,0}$	$132\ (293 + 98\, c_4)$
$w_{3,1}$	$-273\, c_2\ (77 + 265\, c_3 - 122\, c_5)$ $- 14\ (-25872 + 2912\, c_3 + 31614\, c_4 + 16800\, c_5 - 27621\, c_4\, c_5)$ $+ 286\, c_1\ (-32 + 525\, c_4 - 637\, c_6) + 286\ (555 + 112\, c_3)\ c_6$
$w_{3,2}$	$-128557\, c_3 - 147\, c_1\ (-1403 + 1040\, c_2 - 2145\, c_3 + 2640\, c_4 - 572\, c_5)$ $+ 49\ (7623 - 7392\, c_4 + 26\, c_5\ (-32 + 120\, c_2 + 99\, c_5))$ $- 112112\ (c_3 + c_5)\ c_6$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,3}$	$22\,(16464 - 3822\,c_2 + 416\,c_3 - 23511\,c_4) + 210\,c_1\,(-1120 + 1131\,c_2 + 2299\,c_4) + 1274\,(32 + 111\,c_2)\,c_5 + 1001\,(-21 + 47\,c_3 - 86\,c_5)\,c_6$
$w_{3,4}$	$-21\,(-16357 + 18544\,c_3 + 17248\,c_4) - 169598\,c_5 + 7\,\big(-30030\,{c_1}^2 + 52\,c_1\,(416 + 420\,c_2 + 231\,c_5 - 308\,c_6) + 7\,(3120\,c_2\,c_3 + 33\,c_3\,(65\,c_3 + 240\,c_4 - 78\,c_5) + 2288\,c_5\,c_6)\big)$
$w_{3,5}$	$-273\,c_2\,(-77 + 375\,c_3 - 518\,c_5) - 14\,(2912\,c_3 + 297\,c_4\,(66 - 67\,c_5) + 336\,(-77 + 50\,c_5)) - 286\,c_1\,(32 + 525\,c_4 - 413\,c_6) + 286\,(-555 + 112\,c_3)\,c_6$
$w_{3,6}$	$210210\,{c_1}^2 - 11319\,(-33 + 32\,c_4) + 233662\,c_5 + 2548\,c_1\,(-16 + 60\,c_2 - 33\,c_5 - 44\,c_6) + 49\,\big(-2145\,{c_3}^2 + 6\,c_3\,(-344 + 520\,c_2 + 1320\,c_4 + 429\,c_5) + 2288\,c_5\,c_6\big)$
$w_{3,7}$	$22\,(16464 + 3822\,c_2 + 416\,c_3 - 9081\,c_4) - 210\,c_1\,(1120 + 299\,c_2 - 869\,c_4) + 182\,(224 + 183\,c_2)\,c_5 + 1001\,(21 + 17\,c_3 + 22\,c_5)\,c_6$
$w_{3,8}$	$-21\,c_1\,(-13539 + 7280\,c_2 + 15015\,c_3 + 18480\,c_4 + 4004\,c_5) + 7\,(49071 - 51744\,c_4 + 26\,(832 + 840\,c_2 - 693\,c_5)\,c_5) + 143\,(451\,c_3 - 784\,(c_3 + c_5)\,c_6)$
$w_{3,9}$	$158730\,c_2 + 235200\,c_3 - 7\,(12480\,c_2\,c_3 + 33\,c_4\,(1375\,c_3 + 234\,c_5) + 12298\,c_5\,c_6 + 26\,c_1\,(-224 + 345\,c_2 + 1001\,c_6)) + 11\,(-34503\,c_4 + 4\,(-208\,c_5 + 147\,(56 + 13\,c_6)))$
$w_{3,10}$	$-210210\,{c_1}^2 - 49\,(3\,(-2541 + 176\,c_4\,(14 + 15\,c_5) + c_5\,(-1403 + 1040\,c_2 + 858\,c_5)) + 13\,c_3\,(-229 + 240\,c_2 + 33\,c_5 - 176\,c_6)) - 286\,c_1\,(-817 + 735\,c_3 - 392\,c_6)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,11}$	$-16\ (-35362 + 10920\, c_2 + 43197\, c_4 - 4004\, c_6)$
$w_{3,12}$	$210210\, {c_1}^2 - 7\ (3\ (-16357 + (-13539 + 7280\, c_2 - 6006\, c_5)\ c_5 + 1232\, c_4\ (14 + 15\, c_5)) + 13\, c_3\ (2819 + 1680\, c_2 - 231\, c_5 - 1232\, c_6)) + 286\, c_1\ (-593 + 735\, c_3 + 392\, c_6)$
$w_{3,13}$	$362208 + 235200\, c_3 - 390\, c_2\ (407 + 224\, c_3) - 337491\, c_4 - 347655\, c_3\, c_4 - 9152\, c_5 + 54054\, c_4\, c_5 + 2002\ (-42 + 11\, c_5)\ c_6 + 182\, c_1\ (224 + 1305\, c_2 + 649\, c_6)$
$w_{3,14}$	$132\ (-97 + 98\, c_4)$
d_4	$\frac{1}{1441440}$
$w_{4,0}$	$-1848\ (-1 + c_3)$
$w_{4,1}$	$-7\ \left(2288\, {c_1}^2 + 429\, c_2\ (1 + 5\, c_3 + 6\, c_5) - 286\, c_1\ (-8 + 8\, c_3 - 15\, c_4 + 15\, c_6) - 2\ (16\, c_3\ (-257 + 150\, c_5) + 3\ (1232 + 143\, c_4\ (-2 + 3\, c_5) + 40\, c_5\ (-20 + 13\, c_5) + 715\, c_6))\right)$
$w_{4,2}$	$-7\ (-7623 + 11341\, c_3 + 26\ (32 - 111\, c_5)\ c_5 + c_1\ (-4209 + 10813\, c_3 + 6604\, c_5))$
$w_{4,3}$	$14\ \left(3696 - 2400\, c_1 - 858\, c_2 - 2145\, c_1\, c_2 - 2552\, c_3 + 5104\, c_1\, c_3 - 2145\ (1 + c_1)\ c_4 + 13\ (32 + 32\, c_1 + 99\, c_2)\ c_5 + 1560\, {c_5}^2\right) - 3003\ (1 + 5\, c_3 + 6\, c_5)\ c_6$
$w_{4,4}$	$49071 + 16874\, {c_1}^2 - 16874\, c_5 - 52\, c_1\ (-416 + 233\, c_5) + 3\, c_3\ (-29896 + 15125\, c_3 + 1586\, c_5)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,5}$	$-7\left(2288\,c_1{}^2 - 429\,c_2\,(1+5\,c_3+6\,c_5) - 286\,c_1\,(-8+8\,c_3+15\,c_4-15\,c_6) + 2\,(-16\,c_3\,(-257+150\,c_5) + 3\,(-1232+40\,(20-13\,c_5)\,c_5+143\,c_4\,(-2+3\,c_5)+715\,c_6))\right)$
$w_{4,6}$	$7\left(7623 - 3718\,c_1{}^2 + 3718\,c_5 + 52\,c_1\,(-16+127\,c_5) + 3\,c_3\,(-3944+2365\,c_3+962\,c_5)\right)$
$w_{4,7}$	$14\left(3696 - 2552\,c_3 + 2145\,c_4 + 416\,c_5 + 1560\,c_5{}^2 - 429\,c_2\,(-2+3\,c_5) + c_1\,(-2400+2145\,c_2+5104\,c_3+2145\,c_4+416\,c_5)\right) + 3003\,(1+5\,c_3+6\,c_5)\,c_6$
$w_{4,8}$	$49071 - 32197\,c_3 + 26\,c_5\,(832+183\,c_5) + c_1\,(40617-28501\,c_3+12116\,c_5)$
$w_{4,9}$	$-7\left(-4290\,c_2 + 3\,c_3\,(864+2640\,c_3+715\,c_4) + 11\,(39\,c_4+4\,(-168+52\,c_5-39\,c_6)) + 26\left(88\,c_1{}^2 - c_1\,(32+165\,c_2+32\,c_5+165\,c_6) + 3\,c_5\,(40\,c_3+33\,(c_4+c_6))\right)\right)$
$w_{4,10}$	$7\left(7623 - 6791\,c_3 + 3718\,c_1\,(1+c_1+c_3) + 4209\,c_5 - 9981\,c_3\,c_5 - 2886\,c_5{}^2\right)$
$w_{4,11}$	$112\,(853+286\,c_1-957\,c_3-390\,c_5)$
$w_{4,12}$	$49071 - 70703\,c_3 - 16874\,c_1\,(1+c_1+c_3) + 40617\,c_5 - 50133\,c_3\,c_5 - 4758\,c_5{}^2$
$w_{4,13}$	$-7\left(4290\,c_2 + 3\,c_3\,(864+2640\,c_3-715\,c_4) + 11\,(-39\,c_4+4\,(-168+52\,c_5+39\,c_6)) + 26\left(88\,c_1{}^2 + c_1\,(165\,c_2-32\,(1+c_5)+165\,c_6) + 3\,c_5\,(40\,c_3-33\,(c_4+c_6))\right)\right)$
$w_{4,14}$	$-1848\,(-1+c_3)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
d_5	$\dfrac{1}{10090080}$
$w_{5,0}$	$132\ (293 + 98\, c_2)$
$w_{5,1}$	$-286\, c_1\ (32 + 189\, c_2 - 301\, c_4) + 231\, c_2\ (-1643 + 1375\, c_5) - 98\ (-3696 + 416\, c_3 + 858\, c_4 + 1859\, c_3\, c_4 + 2400\, c_5) + 390\ (407 + 161\, c_3 + 224\, c_5)\ c_6$
$w_{5,2}$	$-362208\, c_2 - 286\, c_3\ (817 + 735\, c_3 - 392\, c_4) + 49\ (7623 - 2977\, c_5) - 49\ \big(2574\, {c_1}^2 + 26\, c_5\ (165\, c_3 - 88\, c_4 - 120\, c_6) + 3\, c_1\ (-1403 + 2640\, c_2 - 143\, c_5 + 1040\, c_6)\big)$
$w_{5,3}$	$352\ (1029 + 26\, c_3) - 462\, c_2\ (958 + 325\, c_3) - 286\ (555 + 637\, c_3)\ c_4 + 2912\ (14 + 11\, c_4)\ c_5 + 273\ (-77 + 265\, c_5)\ c_6 + 42\, c_1\ (-5600 + 9207\, c_2 + 793\, c_6)$
$w_{5,4}$	$343497 - 126126\, {c_1}^2 - 284319\, c_3 - 64493\, c_5 + 364\, c_1\ (416 + 231\, c_3 + 308\, c_4 + 420\, c_6) + 49\ (528\, c_2\ (-14 + 15\, c_3) - 143\ (45\, c_3 + 16\, c_4)\ c_5 + 3120\, c_3\, c_6)$
$w_{5,5}$	$286\, c_1\ (-32 + 189\, c_2 - 77\, c_4) + 588\ (616 + 143\, c_4 - 400\, c_5) + 231\, c_2\ (-1461 + 1505\, c_5) + 182\, c_3\ (-224 + 649\, c_4 - 1305\, c_6) + 390\ (-407 + 224\, c_5)\ c_6$
$w_{5,6}$	$-147\ (-2541 + 2464\, c_2 + 1403\, c_3) + 128557\, c_5 + 49\ \big(2574\, {c_1}^2 + 7920\, c_2\, c_3 - 52\, c_1\ (16 + 33\, c_3 - 44\, c_4 - 60\, c_6) + 13\ (495\, c_3\, c_5 - 176\, c_4\, c_5 + 240\, c_3\, c_6)\big)$
$w_{5,7}$	$22\ (16464 + 21\, c_2\ (-594 + 325\, c_3) + 7215\, c_4 + 13\, c_3\ (32 + 413\, c_4)) + 2912\ (14 + 11\, c_4)\ c_5 + 273\ (77 + 375\, c_5)\ c_6 + 42\, c_1\ (-5600 + 6633\, c_2 + 3367\, c_6)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{5,8}$	$343497 - 362208\,c_2 + 286\,c_3\,(593 + 735\,c_3 + 392\,c_4) + 256529\,c_5 + 7\,\big(18018\,{c_1}^2 + 182\,c_5\,(165\,c_3 + 88\,c_4 + 120\,c_6) - 3\,c_1\,(-13539 + 18480\,c_2 + 1001\,c_5 + 7280\,c_6)\big)$
$w_{5,9}$	$-66\,c_2\,(3027 + 2765\,c_3) - 182\,c_1\,(-224 + 121\,c_4 - 183\,c_6) + 210\,c_3\,(1120 + 299\,c_6) + 11\,(91\,c_4\,(-21 + 17\,c_5) + 4\,(8232 - 208\,c_5 + 1911\,c_6))$
$w_{5,10}$	$286\,c_1\,(817 + 294\,c_3 - 392\,c_4 - 441\,c_5) + 49\,\big(7623 + 26\,c_3\,(32 + 165\,c_3 - 88\,c_4) + 2064\,c_5 - 2145\,{c_5}^2 - 528\,c_2\,(14 + 15\,c_5) - 3120\,(c_3 + c_5)\,c_6\big)$
$w_{5,11}$	$-16\,(-35362 + 43197\,c_2 + 4004\,c_4 + 10920\,c_6)$
$w_{5,12}$	$-286\,c_1\,(593 + 294\,c_3 + 392\,c_4 - 441\,c_5) - 7\,(26\,c_3\,(832 + 1155\,c_3 + 616\,c_4) + 3696\,c_2\,(14 + 15\,c_5) - 3\,(16357 + c_5\,(18544 + 5005\,c_5)) + 21840\,(c_3 + c_5)\,c_6)$
$w_{5,13}$	$-66\,c_2\,(7837 + 7315\,c_3) + 210\,c_3\,(1120 - 1131\,c_6) + 182\,c_1\,(224 + 473\,c_4 + 777\,c_6) + 11\,(91\,c_4\,(21 + 47\,c_5) - 4\,(-8232 + 208\,c_5 + 1911\,c_6))$
$w_{5,14}$	$132\,(-97 + 98\,c_2)$
d_6	$\dfrac{1}{10090080}$
$w_{6,0}$	$-12936\,(-1 + c_1)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,1}$	$11\,c_1\,(-24400+9555\,c_2+45472\,c_5) + 49\,\big(429\,c_2\,(-1+6\,c_3) - 2\,\big(1560\,{c_3}^2+13\,c_3\,(32+99\,c_4-32\,c_5) + 429\,c_4\,(2+5\,c_5)+48\,(-77+50\,c_5)\big)+4290\,(1+c_5)\,c_6\big)$
$w_{6,2}$	$-7\,\big(-53361+45375\,{c_1}^2+6\,c_1\,(3268+793\,c_3) + 26\,c_3\,(1001+466\,c_5)+26\,c_5\,(224+649\,c_5)\big)$
$w_{6,3}$	$388080\,{c_1}^2-93808\,c_3-98\,(-3696+2145\,c_4-416\,c_5) - 1274\,((-32\,c_3+165\,c_4-88\,c_5)\,c_5+33\,c_2\,(2+3\,c_3+5\,c_5)) + 21021\,(-1+6\,c_3)\,c_6+147\,c_1\,(-4064+1040\,c_3+715\,c_6)$
$w_{6,4}$	$7\,(3\,c_3\,(-13539+6734\,c_3)+c_1\,(-31729+69867\,c_3+26026\,c_5) - 11\,(-4461+26\,c_5\,(59+91\,c_5)))$
$w_{6,5}$	$-11\,c_1\,(24400+9555\,c_2-45472\,c_5)+49\,\big(-429\,c_2\,(-1+6\,c_3) + 2\,\big(-1560\,{c_3}^2+13\,c_3\,(99\,c_4+32\,(-1+c_5)) + 3\,(1232-800\,c_5+143\,c_4\,(2+5\,c_5)-715\,(1+c_5)\,c_6)\big)\big)$
$w_{6,6}$	$7\,(183\,c_3\,(-161+26\,c_3)+c_1\,(-54895+50133\,c_3-16874\,c_5) + 11\,(4851+26\,c_5\,(91+59\,c_5)))$
$w_{6,7}$	$388080\,{c_1}^2-93808\,c_3+98\,(3696+2145\,c_4+416\,c_5) + 1274\,(33\,c_2\,(2+3\,c_3+5\,c_5)+c_5\,(32\,c_3+165\,c_4+88\,c_5)) + 147\,c_1\,(-4064+1040\,c_3-715\,c_6)-21021\,(-1+6\,c_3)\,c_6$
$w_{6,8}$	$-7\,\big(-49071+49665\,{c_1}^2+6\,c_1\,(2124+3367\,c_3) - 338\,c_5\,(64+77\,c_5)-26\,c_3\,(649+1778\,c_5)\big)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,9}$	$-49\,c_1\,(6560+4800\,c_3-2145\,c_4-2288\,c_5) + 294\,c_3\,(800-520\,c_3+429\,c_4+429\,c_6) + 14014\,(15\,c_2\,(1+c_5)+c_5\,(8\,c_5+15\,c_6)) + 11\,(-1911\,c_4+4\,(8232+2132\,c_5+1911\,c_6))$
$w_{6,10}$	$-7\,(26\,c_3\,(-224+183\,c_3-466\,c_5) - 21\,(2541+1403\,c_5)+11\,c_1\,(2095+2591\,c_5))$
$w_{6,11}$	$-16\,(-28927+46893\,c_1+19110\,c_3+14014\,c_5)$
$w_{6,12}$	$-7\,(26\,c_3\,(832+777\,c_3+1778\,c_5) + 11\,c_1\,(6385+6881\,c_5)-3\,(16357+13539\,c_5))$
$w_{6,13}$	$-49\,c_1\,(6560+4800\,c_3+2145\,c_4-2288\,c_5)-210210\,c_2\,(1+c_5) + 11\,(1911\,c_4+4\,(52\,c_5\,(41+49\,c_5)+147\,(56-13\,c_6))) - 294\,(c_3\,(-800+520\,c_3+429\,c_4)+143\,(3\,c_3+5\,c_5)\,c_6)$
$w_{6,14}$	$-12936\,(-1+c_1)$
d_7	$\dfrac{1}{2522520}$
$w_{7,0}$	9669
$w_{7,1}$	$-4\,(572\,c_1-3003\,c_2+2548\,c_3+42\,(-539+429\,c_4+350\,c_5)-19305\,c_6)$
$w_{7,2}$	$109368\,c_1-110968\,c_3-3626\,(-33+26\,c_5)$
$w_{7,3}$	$-4\,(14700\,c_1+18018\,c_2-572\,c_3+19305\,c_4-7\,(3234+364\,c_5+429\,c_6))$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{7,4}$	$59598 + 121940\,c_1 - 13272\,c_3 - 94952\,c_5$
$w_{7,5}$	$-4\,(572\,c_1 + 7\,(-3234 + 429\,c_2 + 364\,c_3 - 2574\,c_4 + 2100\,c_5) + 19305\,c_6)$
$w_{7,6}$	$-98\,(-1221 + 962\,c_1 + 1116\,c_3) + 110968\,c_5$
$w_{7,7}$	$4\,(22638 - 14700\,c_1 + 18018\,c_2 + 572\,c_3 + 19305\,c_4 + 2548\,c_5 - 3003\,c_6)$
$w_{7,8}$	$59598 + 13272\,c_1 + 94952\,c_3 + 121940\,c_5$
$w_{7,9}$	$4\,(2548\,c_1 + 19305\,c_2 + 14700\,c_3 + 11\,(2058 + 273\,c_4 - 52\,c_5 + 1638\,c_6))$
$w_{7,10}$	$110968\,c_1 + 98\,(1221 + 962\,c_3 + 1116\,c_5)$
$w_{7,11}$	141448
$w_{7,12}$	$-94952\,c_1 - 121940\,c_3 + 42\,(1419 + 316\,c_5)$
$w_{7,13}$	$4\,(2548\,c_1 - 19305\,c_2 + 14700\,c_3 - 11\,(-2058 + 273\,c_4 + 52\,c_5 + 1638\,c_6))$
$w_{7,14}$	-3201
d_8	$\frac{1}{1441440}$
$w_{8,0}$	$1848\,(1 + c_1)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,1}$	$7\left(429\,c_2\,(-1+6\,c_3)+11\,c_1\,(195\,c_2+464\,(1-2\,c_5)) + 2\left(3696+1560\,{c_3}^2-2400\,c_5-429\,c_4\,(2+5\,c_5) - 13\,c_3\,(99\,c_4+32\,(1+c_5))+2145\,(1+c_5)\,c_6\right)\right)$
$w_{8,2}$	$7\left(7623+7095\,{c_1}^2+6\,c_1\,(1972+481\,c_3) - 26\,c_5\,(32+143\,c_5)-26\,c_3\,(143+254\,c_5)\right)$
$w_{8,3}$	$-14\left(3960\,{c_1}^2+24\,c_1\,(-54+65\,c_3)-11\,(336+104\,c_3-195\,c_4) + 13\,(-32+32\,c_3+165\,c_4)\,c_5+1144\,{c_5}^2+429\,c_2\,(2+3\,c_3+5\,c_5)\right) + 3003\,(-1+5\,c_1+6\,c_3)\,c_6$
$w_{8,4}$	$-3\,c_3\,(13539+1586\,c_3)+c_1\,(70703-50133\,c_3+16874\,c_5) - 11\,(-4461+1534\,c_5\,(1+c_5))$
$w_{8,5}$	$-7\left(429\,c_2\,(-1+6\,c_3)+11\,c_1\,(195\,c_2+464\,(-1+2\,c_5)) + 2\left(-1560\,{c_3}^2+c_3\,(-1287\,c_4+416\,(1+c_5)) + 3\,(-1232+800\,c_5-143\,c_4\,(2+5\,c_5)+715\,(1+c_5)\,c_6)\right)\right)$
$w_{8,6}$	$-7\,(3\,c_3\,(1403+962\,c_3)+c_1\,(-6791+9981\,c_3+3718\,c_5) - 11\,(693+338\,c_5\,(1+c_5)))$
$w_{8,7}$	$-7\left(7920\,{c_1}^2-22\,(336+104\,c_3+39\,c_2\,(2+3\,c_3)+195\,c_4) - 26\,(32+165\,c_2-32\,c_3+165\,c_4)\,c_5+2288\,{c_5}^2 + 429\,(-1+6\,c_3)\,c_6+c_1\,(-2592+3120\,c_3+2145\,c_6)\right)$
$w_{8,8}$	$49071+16874\,c_3+3\,c_1\,(29896+15125\,c_1+1586\,c_3) + 21632\,c_5+12116\,c_3\,c_5+16874\,{c_5}^2$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,9}$	$7\,(6\,c_3\,(800 + 520\,c_3 + 429\,c_4) + c_1\,(8224 + 4800\,c_3 + 2145\,c_4 - 2288\,c_5) + 4290\,c_2\,(1 + c_5) + 858\,(3\,c_3 + 5\,c_5)\,c_6 - 11\,(39\,c_4 + 4\,(52\,c_5\,(1 + c_5) - 3\,(56 + 13\,c_6))))$
$w_{8,10}$	$7\,(26\,c_3\,(32 + 111\,c_3 + 254\,c_5) + 11\,c_1\,(1031 + 983\,c_5) + 3\,(2541 + 1403\,c_5))$
$w_{8,11}$	$112\,(853 + 957\,c_1 + 390\,c_3 + 286\,c_5)$
$w_{8,12}$	$26\,c_3\,(-832 + 183\,c_3 - 466\,c_5) + 11\,c_1\,(2927 + 2591\,c_5) + 3\,(16357 + 13539\,c_5)$
$w_{8,13}$	$7\,(6\,c_3\,(800 + 520\,c_3 - 429\,c_4) + c_1\,(8224 + 4800\,c_3 - 2145\,c_4 - 2288\,c_5) - 4290\,c_2\,(1 + c_5) - 858\,(3\,c_3 + 5\,c_5)\,c_6 + 11\,(39\,c_4 - 4\,(-168 + 52\,c_5\,(1 + c_5) + 39\,c_6)))$
$w_{8,14}$	$1848\,(1 + c_1)$
d_9	$\frac{1}{10090080}$
$w_{9,0}$	$132\,(293 - 98\,c_2)$
$w_{9,1}$	$-286\,c_1\,(32 + 189\,c_2 - 77\,c_4) - 231\,c_2\,(-1461 + 1505\,c_5) - 14\,(13\,c_3\,(224 + 649\,c_4) + 42\,(-616 + 143\,c_4 + 400\,c_5)) + 390\,(407 + 609\,c_3 - 224\,c_5)\,c_6$
$w_{9,2}$	$362208\,c_2 - 286\,c_3\,(817 + 735\,c_3 + 392\,c_4) + 49\,(7623 - 2977\,c_5) - 49\,\left(2574\,{c_1}^2 + 26\,c_5\,(165\,c_3 + 88\,c_4 + 120\,c_6) - 3\,c_1\,(1403 + 2640\,c_2 + 143\,c_5 + 1040\,c_6)\right)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{9,3}$	$352\,(1029 + 26\,c_3) - 462\,c_2\,(-594 + 325\,c_3) - 286\,(555 + 413\,c_3)\,c_4 - 2912\,(-14 + 11\,c_4)\,c_5 - 273\,(77 + 375\,c_5)\,c_6 - 42\,c_1\,(5600 + 6633\,c_2 + 3367\,c_6)$
$w_{9,4}$	$343497 - 126126\,{c_1}^2 - 25872\,c_2\,(-14 + 15\,c_3) - 64493\,c_5 + 364\,c_1\,(416 + 231\,c_3 - 308\,c_4 - 420\,c_6) - 7\,(-16016\,c_4\,c_5 + 3\,c_3\,(13539 + 15015\,c_5 + 7280\,c_6))$
$w_{9,5}$	$286\,c_1\,(-32 + 189\,c_2 - 301\,c_4) + 98\,(3696 - 416\,c_3 + 858\,c_4 + 1859\,c_3\,c_4 - 2400\,c_5) - 231\,c_2\,(-1643 + 1375\,c_5) - 390\,(407 + 161\,c_3 + 224\,c_5)\,c_6$
$w_{9,6}$	$147\,(2541 + 2464\,c_2 - 1403\,c_3) + 128557\,c_5 + 49\,\big(2574\,{c_1}^2 - 7920\,c_2\,c_3 - 52\,c_1\,(16 + 33\,c_3 + 44\,c_4 + 60\,c_6) + 13\,(495\,c_3\,c_5 + 176\,c_4\,c_5 - 240\,c_3\,c_6)\big)$
$w_{9,7}$	$22\,(16464 + 416\,c_3 + 21\,c_2\,(958 + 325\,c_3) + 7215\,c_4 + 8281\,c_3\,c_4) - 2912\,(-14 + 11\,c_4)\,c_5 - 273\,(-77 + 265\,c_5)\,c_6 - 42\,c_1\,(5600 + 9207\,c_2 + 793\,c_6)$
$w_{9,8}$	$343497 + 362208\,c_2 + 286\,c_3\,(593 + 735\,c_3 - 392\,c_4) + 256529\,c_5 + 7\,\big(18018\,{c_1}^2 + 182\,c_5\,(165\,c_3 - 88\,c_4 - 120\,c_6) + 3\,c_1\,(13539 + 18480\,c_2 - 1001\,c_5 + 7280\,c_6)\big)$
$w_{9,9}$	$66\,c_2\,(7837 + 7315\,c_3) - 182\,c_1\,(-224 + 473\,c_4 + 777\,c_6) + 210\,c_3\,(1120 + 1131\,c_6) - 11\,(91\,c_4\,(21 + 47\,c_5) + 4\,(208\,c_5 - 147\,(56 + 13\,c_6)))$
$w_{9,10}$	$286\,c_1\,(817 + 294\,c_3 + 392\,c_4 - 441\,c_5) + 49\,\big(7623 + 26\,c_3\,(32 + 165\,c_3 + 88\,c_4) + 2064\,c_5 - 2145\,{c_5}^2 + 528\,c_2\,(14 + 15\,c_5) + 3120\,(c_3 + c_5)\,c_6\big)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{9,11}$	$16\ (35362 + 43197\, c_2 + 4004\, c_4 + 10920\, c_6)$
$w_{9,12}$	$-286\, c_1\ (593 + 294\, c_3 - 392\, c_4 - 441\, c_5) + 7\ (-26\, c_3\ (832 + 1155\, c_3 - 616\, c_4) + 3696\, c_2\ (14 + 15\, c_5) + 3\ (16357 + c_5\ (18544 + 5005\, c_5)) + 21840\ (c_3 + c_5)\ c_6)$
$w_{9,13}$	$66\, c_2\ (3027 + 2765\, c_3) + 210\, c_3\ (1120 - 299\, c_6) + 182\, c_1\ (224 + 121\, c_4 - 183\, c_6) - 11\ (91\, c_4\ (-21 + 17\, c_5) + 4\ (-8232 + 208\, c_5 + 1911\, c_6))$
$w_{9,14}$	$-132\ (97 + 98\, c_2)$
d_{10}	$\dfrac{1}{10090080}$
$w_{10,0}$	$12936\ (1 + c_3)$
$w_{10,1}$	$112112\, {c_1}^2 - 286\, c_1\ (-328 + 392\, c_3 + 735\, c_4 - 735\, c_6) - 49\ (429\, c_2\ (1 + 5\, c_3 + 6\, c_5) + 2\ (-429\, c_4\ (-2 + 3\, c_5) + 120\, c_5\ (20 + 13\, c_5) + 80\, c_3\ (-41 + 30\, c_5) - 33\ (112 + 65\, c_6)))$
$w_{10,2}$	$7\ (53361 + 23045\, c_3 + c_1\ (29463 + 28501\, c_3 - 12116\, c_5) - 26\, c_5\ (224 + 183\, c_5))$
$w_{10,3}$	$-98\, c_1\ (2400 + 2145\, c_2 + 5104\, c_3 + 2145\, c_4 + 416\, c_5) + 2\ \big(134200\, c_3 + 21021\, c_2\ (-2 + 3\, c_5) - 49\ \big(-3696 + 2145\, c_4 - 416\, c_5 + 1560\, {c_5}^2\big)\big) - 21021\ (1 + 5\, c_3 + 6\, c_5)\ c_6$
$w_{10,4}$	$7\ \big(49071 + 21632\, c_1 + 26026\, {c_1}^2 + 12744\, c_3 - 49665\, {c_3}^2 - 26\ (649 + 1778\, c_1 + 777\, c_3)\ c_5\big)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{10,5}$	$112112\,c_1{}^2 - 286\,c_1\,(-328 + 392\,c_3 - 735\,c_4 + 735\,c_6) + 49\,(429\,c_2\,(1 + 5\,c_3 + 6\,c_5) - 2\,(80\,c_3\,(-41 + 30\,c_5) + 3\,(-1232 + 143\,c_4\,(-2 + 3\,c_5) + 40\,c_5\,(20 + 13\,c_5) + 715\,c_6)))$
$w_{10,6}$	$7\,\left(-16874\,c_1{}^2 + 52\,c_1\,(-112 + 233\,c_5) + 77\,(693 + 338\,c_5) - 3\,c_3\,(-6536 + 15125\,c_3 + 1586\,c_5)\right)$
$w_{10,7}$	$98\,c_1\,(-2400 + 2145\,c_2 - 5104\,c_3 + 2145\,c_4 - 416\,c_5) + 2\,(134200\,c_3 - 21021\,c_2\,(-2 + 3\,c_5) + 49\,(3696 + 2145\,c_4 + 104\,(4 - 15\,c_5)\,c_5)) + 21021\,(1 + 5\,c_3 + 6\,c_5)\,c_6$
$w_{10,8}$	$7\,(49071 + 70235\,c_3 + 26\,(832 - 777\,c_5)\,c_5 + c_1\,(40617 + 75691\,c_3 + 46228\,c_5))$
$w_{10,9}$	$210210\,c_2 + 147\,c_3\,(4064 + 2640\,c_3 - 715\,c_4) + 1274\,\left(88\,c_1{}^2 + c_1\,(32 + 165\,c_2 - 32\,c_5 + 165\,c_6) + 3\,c_5\,(40\,c_3 - 33\,(c_4 + c_6))\right) + 11\,(-1911\,c_4 + 4\,(2132\,c_5 + 147\,(56 + 13\,c_6)))$
$w_{10,10}$	$7\,\left(53361 + 54895\,c_3 + 286\,c_1\,(91 + 59\,c_1 + 59\,c_3) + 29463\,c_5 + 50133\,c_3\,c_5 + 4758\,c_5{}^2\right)$
$w_{10,11}$	$16\,(28927 - 14014\,c_1 + 46893\,c_3 + 19110\,c_5)$
$w_{10,12}$	$-7\,\left(-31729\,c_3 + 286\,c_1\,(59 + 91\,c_1 + 91\,c_3) - 69867\,c_3\,c_5 - 20202\,c_5{}^2 - 3\,(16357 + 13539\,c_5)\right)$
$w_{10,13}$	$-210210\,c_2 + 147\,c_3\,(4064 + 2640\,c_3 + 715\,c_4) + 11\,(1911\,c_4 + 4\,(2132\,c_5 + 147\,(56 - 13\,c_6))) + 1274\,\left(88\,c_1{}^2 - c_1\,(-32 + 165\,c_2 + 32\,c_5 + 165\,c_6) + 3\,c_5\,(40\,c_3 + 33\,(c_4 + c_6))\right)$
$w_{10,14}$	$12936\,(1 + c_3)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
d_{11}	$\dfrac{1}{10090080}$
$w_{11,0}$	$132\,(293 - 98\,c_4)$
$w_{11,1}$	$273\,c_2\,(-77 + 375\,c_3 - 518\,c_5) + 286\,c_1\,(-32 + 525\,c_4 - 413\,c_6) - 2912\,c_3\,(14 + 11\,c_6) + 6\,(60368 + 693\,c_4\,(66 - 67\,c_5) - 39200\,c_5 + 26455\,c_6)$
$w_{11,2}$	$-128557\,c_3 + 147\,c_1\,(1403 + 1040\,c_2 + 2145\,c_3 + 2640\,c_4 + 572\,c_5) + 49\,(7623 + 7392\,c_4 + 26\,c_5\,(-32 - 120\,c_2 + 99\,c_5)) + 112112\,(c_3 + c_5)\,c_6$
$w_{11,3}$	$210\,c_1\,(-1120 + 299\,c_2 - 869\,c_4) + 22\,(16464 + 416\,c_3 + 9081\,c_4) + 40768\,c_5 - 546\,c_2\,(154 + 61\,c_5) - 1001\,(21 + 17\,c_3 + 22\,c_5)\,c_6$
$w_{11,4}$	$-210210\,{c_1}^2 + 105105\,{c_3}^2 - 42\,c_3\,(9272 + 3640\,c_2 + 9240\,c_4 + 3003\,c_5) - 364\,c_1\,(-416 + 420\,c_2 - 231\,c_5 - 308\,c_6) + 11\,(31227 + 32928\,c_4 - 26\,c_5\,(593 + 392\,c_6))$
$w_{11,5}$	$273\,c_2\,(77 + 265\,c_3 - 122\,c_5) - 286\,c_1\,(32 + 525\,c_4 - 637\,c_6) - 2912\,c_3\,(14 + 11\,c_6) - 6\,(-154\,(392 + 479\,c_4) + 7\,(5600 + 9207\,c_4)\,c_5 + 26455\,c_6)$
$w_{11,6}$	$210210\,{c_1}^2 + 11319\,(33 + 32\,c_4) + 233662\,c_5 - 2548\,c_1\,(16 + 60\,c_2 + 33\,c_5 - 44\,c_6) + 49\,(-3\,c_3\,(688 + 1040\,c_2 + 715\,c_3 + 2640\,c_4 - 858\,c_5) - 2288\,c_5\,c_6)$
$w_{11,7}$	$-210\,c_1\,(1120 + 1131\,c_2 + 2299\,c_4) + 22\,(16464 + 416\,c_3 + 23511\,c_4) + 40768\,c_5 - 3822\,c_2\,(-22 + 37\,c_5) - 1001\,(-21 + 47\,c_3 - 86\,c_5)\,c_6$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{11,8}$	$21\,c_1\,(13539 + 7280\,c_2 - 15015\,c_3 + 18480\,c_4 - 4004\,c_5) + 7\,(49071 + 51744\,c_4 - 26\,c_5\,(-832 + 840\,c_2 + 693\,c_5)) + 143\,(451\,c_3 + 784\,(c_3 + c_5)\,c_6)$
$w_{11,9}$	$235200\,c_3 + 390\,c_2\,(407 + 224\,c_3) - 182\,c_1\,(-224 + 1305\,c_2 + 649\,c_6) + 77\,(4515\,c_3\,c_4 - 26\,c_5\,(27\,c_4 + 11\,c_6)) + 11\,(30681\,c_4 + 4\,(8232 - 208\,c_5 + 1911\,c_6))$
$w_{11,10}$	$-210210\,{c_1}^2 + 49\,(3\,(2541 + (1403 + 1040\,c_2 - 858\,c_5)\,c_5 + 176\,c_4\,(14 + 15\,c_5)) + 13\,c_3\,(229 + 240\,c_2 - 33\,c_5 - 176\,c_6)) - 286\,c_1\,(-817 + 735\,c_3 + 392\,c_6)$
$w_{11,11}$	$16\,(35362 + 10920\,c_2 + 43197\,c_4 - 4004\,c_6)$
$w_{11,12}$	$210210\,{c_1}^2 + 7\,(3\,(16357 + 1232\,c_4\,(14 + 15\,c_5) + c_5\,(13539 + 7280\,c_2 + 6006\,c_5)) + 13\,c_3\,(-2819 + 1680\,c_2 + 231\,c_5 - 1232\,c_6)) + 286\,c_1\,(-593 + 735\,c_3 - 392\,c_6)$
$w_{11,13}$	$362208 + 235200\,c_3 + 390\,c_2\,(-407 + 224\,c_3) + 379533\,c_4 + 317625\,c_3\,c_4 - 9152\,c_5 + 54054\,c_4\,c_5 + 2002\,(-42 + 43\,c_5)\,c_6 + 182\,c_1\,(224 + 345\,c_2 + 1001\,c_6)$
$w_{11,14}$	$-132\,(97 + 98\,c_4)$
d_{12}	$\frac{1}{1441440}$
$w_{12,0}$	$1848\,(1 + c_5)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{12,1}$	$-7\,(26\,c_1\,(88+99\,c_2+32\,c_3-99\,c_4-120\,c_5) - 429\,c_2\,(-1+5\,c_5)+2\,(13\,c_3\,(32+88\,c_3-165\,c_4+165\,c_6) + 3\,(286\,c_4+24\,c_5\,(-18+55\,c_5)-11\,(112+65\,c_6))))$
$w_{12,2}$	$7\,\big(7623-2886\,{c_1}^2+6791\,c_5+3718\,c_3\,(-1+c_3+c_5)+3\,c_1\,(1403+3327\,c_5)\big)$
$w_{12,3}$	$14\,\big(3696+1560\,{c_1}^2+429\,c_2\,(-2+5\,c_3)-2145\,c_4+4112\,c_5 + 143\,c_3\,(8-8\,c_3+15\,c_4+8\,c_5)+3\,c_1\,(429\,c_2-800\,(1+c_5))\big) - 3003\,(1+6\,c_1-5\,c_5)\,c_6$
$w_{12,4}$	$49071+26\,c_1\,(832+183\,c_1-466\,c_3)-40617\,c_3+32197\,c_5-28501\,c_3\,c_5$
$w_{12,5}$	$7\,(-429\,c_2\,(-1+5\,c_5)+26\,c_1\,(-88+99\,c_2-32\,c_3-99\,c_4+120\,c_5) + 2\,(3\,(1232+286\,c_4+24\,(18-55\,c_5)\,c_5-715\,c_6) - 13\,c_3\,(32+88\,c_3+165\,c_4-165\,c_6)))$
$w_{12,6}$	$7\,(7623-4209\,c_3+26\,c_1\,(-32+111\,c_1+254\,c_3)+11341\,c_5-10813\,c_3\,c_5)$
$w_{12,7}$	$14\,\big(3696+1560\,{c_1}^2-429\,c_2\,(-2+5\,c_3)+2145\,c_4+4112\,c_5 + 143\,c_3\,(8-8\,c_3-15\,c_4+8\,c_5)-3\,c_1\,(429\,c_2+800\,(1+c_5))\big) + 3003\,(1+6\,c_1-5\,c_5)\,c_6$
$w_{12,8}$	$49071-4758\,{c_1}^2+70703\,c_5-16874\,c_3\,(-1+c_3+c_5)+3\,c_1\,(13539+16711\,c_5)$
$w_{12,9}$	$7\,\big(7392+3120\,{c_1}^2-4290\,c_2\,(-1+c_3)+4800\,c_3 - 429\,c_4+5104\,c_5+10208\,c_3\,c_5+2145\,c_4\,c_5 - 858\,(-2+5\,c_3)\,c_6-26\,c_1\,(-32+32\,c_3+99\,c_4+99\,c_6)\big)$
$w_{12,10}$	$-7\,(26\,c_3\,(-32+143\,c_3)+26\,c_1\,(-143+254\,c_3+111\,c_5) - 3\,(2541+c_5\,(3944+2365\,c_5)))$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{12,11}$	$-112\,(-853 + 390\,c_1 + 286\,c_3 - 957\,c_5)$
$w_{12,12}$	$26\,c_3\,(-832 + 649\,c_3) + 26\,c_1\,(-649 + 466\,c_3 - 183\,c_5) + 3\,(16357 + c_5\,(29896 + 15125\,c_5))$
$w_{12,13}$	$7\,\big(7392 + 3120\,{c_1}^2 + 4290\,c_2\,(-1 + c_3) + 4800\,c_3 + 429\,c_4 + 5104\,c_5 + 10208\,c_3\,c_5 - 2145\,c_4\,c_5 - 26\,c_1\,(-32 + 32\,c_3 - 99\,c_4 - 99\,c_6) + 858\,(-2 + 5\,c_3)\,c_6\big)$
$w_{12,14}$	$1848\,(1 + c_5)$
d_{13}	$\dfrac{1}{10090080}$
$w_{13,0}$	$132\,(293 - 98\,c_6)$
$w_{13,1}$	$-143\,c_1\,(64 + 329\,c_2) + 1001\,c_2\,(-21 + 86\,c_3) - 196\,(-1848 + 208\,c_3 + 429\,c_4 + 1200\,c_5) + 517242\,c_6 - 42\,(3367\,c_3\,c_4 + 5\,c_5\,(-1131\,c_4 + 2299\,c_6))$
$w_{13,2}$	$-105105\,{c_1}^2 - 286\,c_3\,(817 + 392\,c_2 - 294\,c_5) + 294\,c_1\,(344 + 429\,c_3 - 520\,c_4 + 1320\,c_6) + 49\,(26\,c_5\,(-32 - 88\,c_2 + 120\,c_4 + 165\,c_5) + 231\,(33 + 32\,c_6))$
$w_{13,3}$	$-2002\,c_2\,(42 + 11\,c_3 - 59\,c_5) + 1568\,(231 + 26\,c_5) + 390\,c_4\,(-407 + 609\,c_5) + 337491\,c_6 + 286\,c_3\,(32 + 189\,c_6) + 105\,c_1\,(832\,c_4 - 7\,(320 + 473\,c_6))$
$w_{13,4}$	$-91\,c_1\,(-2819 + 1232\,c_2 - 231\,c_3 - 1680\,c_4 + 2310\,c_5) + 21\,c_3\,(-13539 + 6006\,c_3 + 7280\,c_4 - 18480\,c_6) + 11\,(26\,c_5\,(-593 + 392\,c_2 + 735\,c_5) + 21\,(1487 + 1568\,c_6))$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{13,5}$	$-143\,c_1\,(64+119\,c_2) - 196\,(-1848+208\,c_3-429\,c_4+1200\,c_5)+199782\,c_6 - 7\,(143\,c_2\,(-21+22\,c_3)+78\,c_4\,(61\,c_3+115\,c_5)+26070\,c_5\,c_6)$
$w_{13,6}$	$-637\,c_1\,(229+176\,c_2+33\,c_3-240\,c_4-330\,c_5) - 147\,c_3\,(1403+858\,c_3-1040\,c_4+2640\,c_6) + 11\,(26\,(817+392\,c_2-735\,c_5)\,c_5+1029\,(33+32\,c_6))$
$w_{13,7}$	$2002\,c_2\,(42+43\,c_3-91\,c_5)+1568\,(231+26\,c_5) - 390\,c_4\,(-407+161\,c_5)+105\,c_1\,(-2240+832\,c_4-3025\,c_6) + 379533\,c_6-286\,c_3\,(-32+189\,c_6)$
$w_{13,8}$	$169598\,c_3 + 7\,\big(15015\,{c_1}^2-182\,(88\,c_2\,(c_3+c_5)+3\,c_5\,(22\,c_3-40\,c_4+55\,c_5)) - 6\,c_1\,(-9272+3003\,c_3+3640\,c_4-9240\,c_6)\big) + 7\,(21632\,c_5+33\,(1487+1568\,c_6))$
$w_{13,9}$	$362208+235200\,c_3-91\,c_1\,(-448+352\,c_2-795\,c_4) - 21021\,c_4-33306\,c_3\,c_4-9152\,c_5 - 286\,c_2\,(-555+637\,c_5)+462\,(958+837\,c_3-325\,c_5)\,c_6$
$w_{13,10}$	$143\,c_1\,(899+784\,c_2-2205\,c_5) + 49\,\big(2574\,{c_3}^2+52\,c_3\,(16+44\,c_2-60\,c_4-33\,c_5) + 3\,(2541+1403\,c_5-1040\,c_4\,c_5+176\,(14+15\,c_5)\,c_6)\big)$
$w_{13,11}$	$16\,(35362+4004\,c_2-10920\,c_4+43197\,c_6)$
$w_{13,12}$	$143\,c_1\,(-451+784\,c_2+2205\,c_5) + 7\,\big(-18018\,{c_3}^2+52\,c_3\,(-416+308\,c_2-420\,c_4+231\,c_5) + 3\,(16357+13539\,c_5-7280\,c_4\,c_5+1232\,(14+15\,c_5)\,c_6)\big)$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{13,13}$	$362208 + 235200\,c_3 - 91\,c_1\,(-448 + 352\,c_2 - 1125\,c_4) + 21021\,c_4 - 141414\,c_3\,c_4 - 9152\,c_5 + 286\,c_2\,(-555 + 413\,c_5) + 462\,(594 + 603\,c_3 + 325\,c_5)\,c_6$
$w_{13,14}$	$-132\,(97 + 98\,c_6)$
d_{14}	$\frac{1}{630630}$
$w_{14,0}$	1617
$w_{14,1}$	$-4\,(286\,c_1 + 49\,(-231 + 26\,c_3 + 150\,c_5))$
$w_{14,2}$	$14\,(3201 + 2190\,c_1 - 286\,c_3 + 494\,c_5)$
$w_{14,3}$	$4\,(-7350\,c_1 + 286\,c_3 + 49\,(231 + 26\,c_5))$
$w_{14,4}$	$14\,(3201 + 494\,c_1 - 2190\,c_3 + 286\,c_5)$
$w_{14,5}$	$-4\,(286\,c_1 + 49\,(-231 + 26\,c_3 + 150\,c_5))$
$w_{14,6}$	$14\,(3201 + 494\,c_1 - 2190\,c_3 + 286\,c_5)$
$w_{14,7}$	$4\,(-7350\,c_1 + 286\,c_3 + 49\,(231 + 26\,c_5))$
$w_{14,8}$	$14\,(3201 + 2190\,c_1 - 286\,c_3 + 494\,c_5)$
$w_{14,9}$	$196\,(231 + 26\,c_1 + 150\,c_3) - 1144\,c_5$

(*continues*)

Table B.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{14,10}$	$14\ (3201 + 286\, c_1 - 494\, c_3 + 2190\, c_5)$
$w_{14,11}$	70724
$w_{14,12}$	$14\ (3201 + 286\, c_1 - 494\, c_3 + 2190\, c_5)$
$w_{14,13}$	$196\ (231 + 26\, c_1 + 150\, c_3) - 1144\, c_5$
$w_{14,14}$	1617

Appendix C

Coefficients for second-order methods

In the following tables, exact coefficients for a family of implicit Chebyshev methods for the numerical integration of second-order differential equations are given. These coefficients for even degrees $2 \leq n \leq 14$ are used to recast Eq. (1.11), such that

$$y\left(x + h\xi_j\right) \; = 2y(x) + y\left(x - h\xi_j\right) + I_j, \tag{C.1}$$

where

$$I_j \; = h^2 \, d_j \sum_{k=0}^{n} w_{jk} \, \left(f(x_k^+) + f(x_k^-)\right), \tag{C.2}$$

where d_j represents a common denominator for each expression.

The constants c_ℓ appearing in Table C.7 are defined in Appendix A, Table A.8.

Table C.1: Integral approximation I_j for $n = 2$.

d_j, w_{jk}	Expression	w_{jk}	Expression
d_1	$\frac{1}{96}$	$w_{1,1}$	6
$w_{1,0}$	14	$w_{1,2}$	-1
d_2	$\frac{1}{3}$	$w_{2,1}$	1
$w_{2,0}$	1	$w_{2,2}$	0

Table C.2: Integral approximation I_j for $n = 4$.

d_j, w_{jk}	Expression	w_{jk}	Expression
d_1	$\frac{1}{1920}$	$w_{1,2}$	$134 - 96\sqrt{2}$
$w_{1,0}$	$66 - 30\sqrt{2}$	$w_{1,3}$	$182 - 128\sqrt{2}$
$w_{1,1}$	10	$w_{1,4}$	$1 - \sqrt{2}$
d_2	$\frac{1}{480}$	$w_{2,2}$	10
$w_{2,0}$	18	$w_{2,3}$	$20 - 16\sqrt{2}$
$w_{2,1}$	$4\left(5 + 4\sqrt{2}\right)$	$w_{2,4}$	1
d_3	$\frac{1}{1920}$	$w_{3,2}$	$134 + 96\sqrt{2}$
$w_{3,0}$	$6\left(11 + 5\sqrt{2}\right)$	$w_{3,3}$	10
$w_{3,1}$	$182 + 128\sqrt{2}$	$w_{3,4}$	$1 + \sqrt{2}$
d_4	$\frac{1}{15}$	$w_{4,2}$	3
$w_{4,0}$	1	$w_{4,3}$	$2 - \sqrt{2}$
$w_{4,1}$	$2 + \sqrt{2}$	$w_{4,4}$	0

Table C.3: Integral approximation I_j for $n = 6$.

d_j, w_{jk}	Expression	w_{jk}	Expression
d_1	$\frac{1}{322560}$	$w_{1,3}$	$18206 - 10496\sqrt{3}$
$w_{1,0}$	$4706 - 2240\sqrt{3}$	$w_{1,4}$	$6710 - 3884\sqrt{3}$
$w_{1,1}$	350	$w_{1,5}$	$17750 - 10240\sqrt{3}$
$w_{1,2}$	$25142 - 14548\sqrt{3}$	$w_{1,6}$	$49 - 32\sqrt{3}$
d_2	$\frac{1}{322560}$	$w_{2,3}$	-418
$w_{2,0}$	2466	$w_{2,4}$	234
$w_{2,1}$	$3530 + 2140\sqrt{3}$	$w_{2,5}$	$3530 - 2140\sqrt{3}$
$w_{2,2}$	1890	$w_{2,6}$	81
d_3	$\frac{1}{10080}$	$w_{3,3}$	70
$w_{3,0}$	142	$w_{3,4}$	-8
$w_{3,1}$	$40\left(7 + 4\sqrt{3}\right)$	$w_{3,5}$	$40\left(7 - 4\sqrt{3}\right)$
$w_{3,2}$	568	$w_{3,6}$	-1
d_4	$\frac{1}{35840}$	$w_{4,3}$	2286
$w_{4,0}$	786	$w_{4,4}$	210

(continues)

Table C.3: Integral approximation I_j, $n = 6$. (*continued*)

d_j, w_{jk}	Expression	w_{jk}	Expression
$w_{4,1}$	$90\left(17+10\sqrt{3}\right)$	$w_{4,5}$	$3\left(510-300\sqrt{3}\right)$
$w_{4,2}$	4122	$w_{4,6}$	9
d_5	$\frac{1}{322560}$	$w_{5,3}$	$18206+10496\sqrt{3}$
$w_{5,0}$	$4706+2240\sqrt{3}$	$w_{5,4}$	$6710+3884\sqrt{3}$
$w_{5,1}$	$17750+10240\sqrt{3}$	$w_{5,5}$	350
$w_{5,2}$	$25142+14548\sqrt{3}$	$w_{5,6}$	$49+32\sqrt{3}$
d_6	$\frac{1}{315}$	$w_{6,3}$	41
$w_{6,0}$	9	$w_{6,4}$	18
$w_{6,1}$	$10\left(2+\sqrt{3}\right)$	$w_{6,5}$	$-10\left(-2+\sqrt{3}\right)$
$w_{6,2}$	54	$w_{6,6}$	0

Table C.4: Integral approximation I_j for $n = 8$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{161280}$
$w_{1,0}$	$2\left(643+4\sqrt{2}-315\sqrt{2+\sqrt{2}}\right)$

(*continues*)

Table C.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,1}$	$-6\left(-49+28\sqrt{2}+128\sqrt{2-\sqrt{2}}+128\sqrt{2+\sqrt{2}}-128\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,2}$	$3256\sqrt{2}+168\sqrt{4-2\sqrt{2}}-11\left(-416+256\sqrt{2+\sqrt{2}}+145\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,3}$	$30\sqrt{2}+4\left(977+448\sqrt{2-\sqrt{2}}-448\sqrt{2+\sqrt{2}}-192\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,4}$	$5146+1544\sqrt{2}-3968\sqrt{2+\sqrt{2}}$
$w_{1,5}$	$4\left(1391+701\sqrt{2}-448\sqrt{2-\sqrt{2}}-832\sqrt{2+\sqrt{2}}-192\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,6}$	$1595\sqrt{2\left(2+\sqrt{2}\right)}-8\left(-572+297\sqrt{2}+21\sqrt{4-2\sqrt{2}}+352\sqrt{2+\sqrt{2}}\right)$
$w_{1,7}$	$2\left(2765-29\sqrt{2}+384\sqrt{2-\sqrt{2}}-2176\sqrt{2+\sqrt{2}}+384\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{1,8}$	$3+4\sqrt{2}-5\sqrt{2+\sqrt{2}}$
d_2	$\frac{1}{40320}$
$w_{2,0}$	$324-160\sqrt{2}$
$w_{2,1}$	$1360-853\sqrt{2}+159\sqrt{4-2\sqrt{2}}-192\sqrt{2-\sqrt{2}}$ $+448\sqrt{2+\sqrt{2}}-266\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{2,2}$	84

(*continues*)

Table C.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,3}$	$592 - 427\sqrt{2} - 266\sqrt{4-2\sqrt{2}} + 448\sqrt{2-\sqrt{2}} + 192\sqrt{2+\sqrt{2}} - 159\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{2,4}$	$1412 - 992\sqrt{2}$
$w_{2,5}$	$592 - 427\sqrt{2} + 266\sqrt{4-2\sqrt{2}} - 448\sqrt{2-\sqrt{2}} - 192\sqrt{2+\sqrt{2}} + 159\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{2,6}$	$1996 - 1408\sqrt{2}$
$w_{2,7}$	$1360 - 853\sqrt{2} - 159\sqrt{4-2\sqrt{2}} + 192\sqrt{2-\sqrt{2}} - 448\sqrt{2+\sqrt{2}} + 266\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{2,8}$	2
d_3	$\frac{1}{161280}$
$w_{3,0}$	$-2\left(-643 + 4\sqrt{2} + 315\sqrt{2-\sqrt{2}}\right)$
$w_{3,1}$	$4\left(1391 - 701\sqrt{2} + 192\sqrt{4-2\sqrt{2}} - 832\sqrt{2-\sqrt{2}} + 448\sqrt{2+\sqrt{2}}\right)$
$w_{3,2}$	$-168\sqrt{2\left(2+\sqrt{2}\right)} + 11\left(416 + 216\sqrt{2} - 145\sqrt{4-2\sqrt{2}} - 256\sqrt{2-\sqrt{2}}\right)$
$w_{3,3}$	$6\left(49 + 28\sqrt{2} - 128\sqrt{4-2\sqrt{2}} - 128\sqrt{2-\sqrt{2}} + 128\sqrt{2+\sqrt{2}}\right)$

(*continues*)

Table C.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,4}$	$5146 - 1544\sqrt{2} - 3968\sqrt{2-\sqrt{2}}$
$w_{3,5}$	$2\left(2765 + 29\sqrt{2} - 384\sqrt{4-2\sqrt{2}} - 2176\sqrt{2-\sqrt{2}} - 384\sqrt{2+\sqrt{2}}\right)$
$w_{3,6}$	$4576 - 3256\sqrt{2} + 1595\sqrt{4-2\sqrt{2}} - 2816\sqrt{2-\sqrt{2}} + 168\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{3,7}$	$-30\sqrt{2} + 4\left(977 + 192\sqrt{4-2\sqrt{2}} - 448\sqrt{2-\sqrt{2}} - 448\sqrt{2+\sqrt{2}}\right)$
$w_{3,8}$	$3 - 4\sqrt{2} - 5\sqrt{2-\sqrt{2}}$
d_4	$\dfrac{1}{10080}$
$w_{4,0}$	82
$w_{4,1}$	$-48\sqrt{2-\sqrt{2}} + 56\left(3 + 2\sqrt{2+\sqrt{2}}\right)$
$w_{4,2}$	$4\left(63 + 44\sqrt{2}\right)$
$w_{4,3}$	$8\left(21 + 14\sqrt{2-\sqrt{2}} + 6\sqrt{2+\sqrt{2}}\right)$
$w_{4,4}$	42
$w_{4,5}$	$-8\left(-21 + 14\sqrt{2-\sqrt{2}} + 6\sqrt{2+\sqrt{2}}\right)$
$w_{4,6}$	$252 - 176\sqrt{2}$

(*continues*)

Table C.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,7}$	$8\left(21+6\sqrt{2-\sqrt{2}}-14\sqrt{2+\sqrt{2}}\right)$
$w_{4,8}$	1
d_5	$\frac{1}{161280}$
$w_{5,0}$	$1286-8\sqrt{2}+630\sqrt{2-\sqrt{2}}$
$w_{5,1}$	$-30\sqrt{2}+4\left(977-192\sqrt{4-2\sqrt{2}}+448\sqrt{2-\sqrt{2}}+448\sqrt{2+\sqrt{2}}\right)$
$w_{5,2}$	$168\sqrt{2\left(2+\sqrt{2}\right)}+11\left(416+216\sqrt{2}+145\sqrt{4-2\sqrt{2}}+256\sqrt{2-\sqrt{2}}\right)$
$w_{5,3}$	$2\left(2765+29\sqrt{2}+384\sqrt{4-2\sqrt{2}}+2176\sqrt{2-\sqrt{2}}+384\sqrt{2+\sqrt{2}}\right)$
$w_{5,4}$	$5146-1544\sqrt{2}+3968\sqrt{2-\sqrt{2}}$
$w_{5,5}$	$6\left(49+28\sqrt{2}+128\sqrt{4-2\sqrt{2}}+128\sqrt{2-\sqrt{2}}-128\sqrt{2+\sqrt{2}}\right)$
$w_{5,6}$	$4576-3256\sqrt{2}-1595\sqrt{4-2\sqrt{2}}+2816\sqrt{2-\sqrt{2}}-168\sqrt{2\left(2+\sqrt{2}\right)}$
$w_{5,7}$	$-4\left(-1391+701\sqrt{2}+192\sqrt{4-2\sqrt{2}}-832\sqrt{2-\sqrt{2}}+448\sqrt{2+\sqrt{2}}\right)$
$w_{5,8}$	$3-4\sqrt{2}+5\sqrt{2-\sqrt{2}}$
d_6	$\frac{1}{40320}$

(*continues*)

Table C.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,0}$	$324 + 160\sqrt{2}$
$w_{6,1}$	$592 + 427\sqrt{2} - 159\sqrt{4 - 2\sqrt{2}} - 192\sqrt{2 - \sqrt{2}} + 448\sqrt{2 + \sqrt{2}} + 266\sqrt{2\left(2 + \sqrt{2}\right)}$
$w_{6,2}$	$4\left(499 + 352\sqrt{2}\right)$
$w_{6,3}$	$1360 + 853\sqrt{2} + 266\sqrt{4 - 2\sqrt{2}} + 448\sqrt{2 - \sqrt{2}} + 192\sqrt{2 + \sqrt{2}} + 159\sqrt{2\left(2 + \sqrt{2}\right)}$
$w_{6,4}$	$4\left(353 + 248\sqrt{2}\right)$
$w_{6,5}$	$1360 + 853\sqrt{2} - 266\sqrt{4 - 2\sqrt{2}} - 448\sqrt{2 - \sqrt{2}} - 192\sqrt{2 + \sqrt{2}} - 159\sqrt{2\left(2 + \sqrt{2}\right)}$
$w_{6,6}$	84
$w_{6,7}$	$592 + 427\sqrt{2} + 159\sqrt{4 - 2\sqrt{2}} + 192\sqrt{2 - \sqrt{2}} - 448\sqrt{2 + \sqrt{2}} - 266\sqrt{2\left(2 + \sqrt{2}\right)}$
$w_{6,8}$	2
d_7	$\frac{1}{161280}$
$w_{7,0}$	$2\left(643 + 4\sqrt{2} + 315\sqrt{2 + \sqrt{2}}\right)$

(*continues*)

Table C.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{7,1}$	$-2\left(-2765+29\sqrt{2}+384\sqrt{2-\sqrt{2}}-2176\sqrt{2+\sqrt{2}}+384\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,2}$	$3256\sqrt{2}-168\sqrt{4-2\sqrt{2}}+11\left(416+256\sqrt{2+\sqrt{2}}+145\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,3}$	$4\left(1391+701\sqrt{2}+448\sqrt{2-\sqrt{2}}+832\sqrt{2+\sqrt{2}}+192\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,4}$	$5146+1544\sqrt{2}+3968\sqrt{2+\sqrt{2}}$
$w_{7,5}$	$30\sqrt{2}+4\left(977-448\sqrt{2-\sqrt{2}}+448\sqrt{2+\sqrt{2}}+192\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,6}$	$-2376\sqrt{2}+168\sqrt{4-2\sqrt{2}}+11\left(416+256\sqrt{2+\sqrt{2}}-145\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,7}$	$-6\left(-49+28\sqrt{2}-128\sqrt{2-\sqrt{2}}-128\sqrt{2+\sqrt{2}}+128\sqrt{2\left(2+\sqrt{2}\right)}\right)$
$w_{7,8}$	$3+4\sqrt{2}+5\sqrt{2+\sqrt{2}}$
d_8	$\frac{1}{315}$
$w_{8,0}$	5
$w_{8,1}$	$20-6\sqrt{2}-3\sqrt{2-\sqrt{2}}+7\sqrt{2+\sqrt{2}}$
$w_{8,2}$	$11\left(2+\sqrt{2}\right)$
$w_{8,3}$	$20+6\sqrt{2}+7\sqrt{2-\sqrt{2}}+3\sqrt{2+\sqrt{2}}$

(*continues*)

Table C.4: Integral approximation I_j, $n = 8$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,4}$	31
$w_{8,5}$	$20 + 6\sqrt{2} - 7\sqrt{2-\sqrt{2}} - 3\sqrt{2+\sqrt{2}}$
$w_{8,6}$	$-11\left(-2+\sqrt{2}\right)$
$w_{8,7}$	$20 - 6\sqrt{2} + 3\sqrt{2-\sqrt{2}} - 7\sqrt{2+\sqrt{2}}$
$w_{8,8}$	0

Table C.5: Integral approximation I_j for $n = 10$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{35481600}$
$w_{1,0}$	$14\left(12861 + 25\sqrt{5} - 3168\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{1,1}$	$-154\left(-201 + 75\sqrt{5} + 512\sqrt{10-2\sqrt{5}} + 256\sqrt{2\left(5+\sqrt{5}\right)} - 256\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{1,2}$	$2\left(453362 + 121188\sqrt{5} + 4950\sqrt{50-10\sqrt{5}} - 4125\sqrt{10-2\sqrt{5}} - 148694\sqrt{2\left(5+\sqrt{5}\right)} - 20619\sqrt{10\left(5+\sqrt{5}\right)}\right)$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,3}$	$-4\left(-131931+13283\sqrt{5}-38400\sqrt{10-2\sqrt{5}} + 28544\sqrt{2\left(5+\sqrt{5}\right)}+9856\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{1,4}$	$834974+374354\sqrt{5}+9900\sqrt{50-10\sqrt{5}}+8250\sqrt{10-2\sqrt{5}} - 240212\sqrt{2\left(5+\sqrt{5}\right)}-97558\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,5}$	$6\left(178759+18715\sqrt{5}-57984\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{1,6}$	$-2\left(-545487+53463\sqrt{5}+4950\sqrt{50-10\sqrt{5}}+4125\sqrt{10-2\sqrt{5}} + 148694\sqrt{2\left(5+\sqrt{5}\right)}-20619\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{1,7}$	$4\left(235331+88271\sqrt{5}-38400\sqrt{10-2\sqrt{5}} - 67968\sqrt{2\left(5+\sqrt{5}\right)}-9856\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{1,8}$	$650724-238904\sqrt{5}-9900\sqrt{50-10\sqrt{5}}+8250\sqrt{10-2\sqrt{5}} - 240212\sqrt{2\left(5+\sqrt{5}\right)}+97558\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,9}$	$1060794-117542\sqrt{5}+78848\sqrt{10-2\sqrt{5}} - 346624\sqrt{2\left(5+\sqrt{5}\right)}+39424\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{1,10}$	$7\left(61+25\sqrt{5}-32\sqrt{2\left(5+\sqrt{5}\right)}\right)$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
d_2	$\frac{1}{35481600}$
$w_{2,0}$	$-350\left(-389+129\sqrt{5}\right)$
$w_{2,1}$	$529692-211000\sqrt{5}+30800\sqrt{50-10\sqrt{5}}-61446\sqrt{10-2\sqrt{5}}$ $+115200\sqrt{2\left(5+\sqrt{5}\right)}-46650\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{2,2}$	$-11550\left(-5+\sqrt{5}\right)$
$w_{2,3}$	$260962-118446\sqrt{5}-46650\sqrt{50-10\sqrt{5}}+115200\sqrt{10-2\sqrt{5}}$ $+61446\sqrt{2\left(5+\sqrt{5}\right)}-30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{2,4}$	$3800-240\sqrt{5}$
$w_{2,5}$	$422442-189858\sqrt{5}$
$w_{2,6}$	$-400\left(-2307+1030\sqrt{5}\right)$
$w_{2,7}$	$260962-118446\sqrt{5}+46650\sqrt{50-10\sqrt{5}}-115200\sqrt{10-2\sqrt{5}}$ $-61446\sqrt{2\left(5+\sqrt{5}\right)}+30800\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{2,8}$	$823750-367910\sqrt{5}$
$w_{2,9}$	$-211000\sqrt{5}-30800\sqrt{50-10\sqrt{5}}+6\left(88282+10241\sqrt{10-2\sqrt{5}}\right.$ $\left.-19200\sqrt{2\left(5+\sqrt{5}\right)}+7775\sqrt{10\left(5+\sqrt{5}\right)}\right)$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,10}$	$-175\left(-5+\sqrt{5}\right)$
d_3	$\frac{1}{35481600}$
$w_{3,0}$	$-14\left(-12861+25\sqrt{5}+3168\sqrt{10-2\sqrt{5}}\right)$
$w_{3,1}$	$4\left(235331-88271\sqrt{5}+9856\sqrt{50-10\sqrt{5}} - 67968\sqrt{10-2\sqrt{5}}+38400\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{3,2}$	$2\left(545487+53463\sqrt{5}-20619\sqrt{50-10\sqrt{5}}-148694\sqrt{10-2\sqrt{5}} + 4125\sqrt{2\left(5+\sqrt{5}\right)}-4950\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{3,3}$	$154\left(201+75\sqrt{5}-256\sqrt{50-10\sqrt{5}} - 256\sqrt{10-2\sqrt{5}}+512\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{3,4}$	$650724+238904\sqrt{5}-97558\sqrt{50-10\sqrt{5}}-240212\sqrt{10-2\sqrt{5}} - 8250\sqrt{2\left(5+\sqrt{5}\right)}-9900\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,5}$	$-6\left(-178759+18715\sqrt{5}+57984\sqrt{10-2\sqrt{5}}\right)$
$w_{3,6}$	$906724-242376\sqrt{5}+41238\sqrt{50-10\sqrt{5}}-297388\sqrt{10-2\sqrt{5}} + 8250\sqrt{2\left(5+\sqrt{5}\right)}+9900\sqrt{10\left(5+\sqrt{5}\right)}$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,7}$	$2\left(530397 + 58771\sqrt{5} - 19712\sqrt{50-10\sqrt{5}} - 173312\sqrt{10-2\sqrt{5}} - 39424\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{3,8}$	$834974 - 374354\sqrt{5} + 97558\sqrt{50-10\sqrt{5}} - 240212\sqrt{10-2\sqrt{5}} - 8250\sqrt{2\left(5+\sqrt{5}\right)} + 9900\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{3,9}$	$4\left(131931 + 13283\sqrt{5} + 9856\sqrt{50-10\sqrt{5}} - 28544\sqrt{10-2\sqrt{5}} - 38400\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{3,10}$	$-7\left(-61 + 25\sqrt{5} + 32\sqrt{10-2\sqrt{5}}\right)$
d_4	$\frac{1}{1419264}$
$w_{4,0}$	$9030 - 1778\sqrt{5}$
$w_{4,1}$	$41650 - 13858\sqrt{5} + 1232\sqrt{50-10\sqrt{5}} - 3850\sqrt{10-2\sqrt{5}} + 7680\sqrt{2\left(5+\sqrt{5}\right)} - 1866\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{4,2}$	$43160 - 4496\sqrt{5}$
$w_{4,3}$	$20860 - 3848\sqrt{5} - 1866\sqrt{50-10\sqrt{5}} + 7680\sqrt{10-2\sqrt{5}} + 3850\sqrt{2\left(5+\sqrt{5}\right)} - 1232\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{4,4}$	$462\left(5+\sqrt{5}\right)$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,5}$	$44730 - 20238\sqrt{5}$
$w_{4,6}$	$53430 - 23786\sqrt{5}$
$w_{4,7}$	$20860 - 3848\sqrt{5} + 1866\sqrt{50 - 10\sqrt{5}} - 7680\sqrt{10 - 2\sqrt{5}} - 3850\sqrt{2\left(5 + \sqrt{5}\right)} + 1232\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{4,8}$	$59440 - 26528\sqrt{5}$
$w_{4,9}$	$41650 - 13858\sqrt{5} - 1232\sqrt{50 - 10\sqrt{5}} + 3850\sqrt{10 - 2\sqrt{5}} - 7680\sqrt{2\left(5 + \sqrt{5}\right)} + 1866\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{4,10}$	$7\left(5 + \sqrt{5}\right)$
d_5	$\frac{1}{554400}$
$w_{5,0}$	2786
$w_{5,1}$	$2400\sqrt{2\left(5 + \sqrt{5}\right)} - 308\left(-27 + 3\sqrt{5} + 4\sqrt{10 - 2\sqrt{5}}\right)$
$w_{5,2}$	$4\left(2579 + 1171\sqrt{5}\right)$
$w_{5,3}$	$4\left(2079 + 231\sqrt{5} + 600\sqrt{10 - 2\sqrt{5}} + 308\sqrt{2\left(5 + \sqrt{5}\right)}\right)$
$w_{5,4}$	$6316 + 2836\sqrt{5}$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{5,5}$	1386
$w_{5,6}$	$10316 - 4684\sqrt{5}$
$w_{5,7}$	$4\left(2079 + 231\sqrt{5} - 600\sqrt{10 - 2\sqrt{5}} - 308\sqrt{2\left(5 + \sqrt{5}\right)}\right)$
$w_{5,8}$	$6316 - 2836\sqrt{5}$
$w_{5,9}$	$-2400\sqrt{2\left(5 + \sqrt{5}\right)} - 308\left(-27 + 3\sqrt{5} - 4\sqrt{10 - 2\sqrt{5}}\right)$
$w_{5,10}$	-7
d_6	$\frac{1}{35481600}$
$w_{6,0}$	$350\left(389 + 129\sqrt{5}\right)$
$w_{6,1}$	$2\left(130481 + 59223\sqrt{5} - 15400\sqrt{50 - 10\sqrt{5}} - 30723\sqrt{10 - 2\sqrt{5}} + 57600\sqrt{2\left(5 + \sqrt{5}\right)} + 23325\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{6,2}$	$400\left(2307 + 1030\sqrt{5}\right)$
$w_{6,3}$	$2\left(264846 + 105500\sqrt{5} + 23325\sqrt{50 - 10\sqrt{5}} + 57600\sqrt{10 - 2\sqrt{5}} + 30723\sqrt{2\left(5 + \sqrt{5}\right)} + 15400\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{6,4}$	$823750 + 367910\sqrt{5}$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,5}$	$6\left(70407 + 31643\sqrt{5}\right)$
$w_{6,6}$	$11550\left(5 + \sqrt{5}\right)$
$w_{6,7}$	$2\left(264846 + 105500\sqrt{5} - 23325\sqrt{50 - 10\sqrt{5}} - 57600\sqrt{10 - 2\sqrt{5}} - 30723\sqrt{2\left(5 + \sqrt{5}\right)} - 15400\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{6,8}$	$40\left(95 + 6\sqrt{5}\right)$
$w_{6,9}$	$2\left(130481 + 59223\sqrt{5} + 15400\sqrt{50 - 10\sqrt{5}} + 30723\sqrt{10 - 2\sqrt{5}} - 57600\sqrt{2\left(5 + \sqrt{5}\right)} - 23325\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{6,10}$	$175\left(5 + \sqrt{5}\right)$
d_7	$\dfrac{1}{35481600}$
$w_{7,0}$	$14\left(12861 - 25\sqrt{5} + 3168\sqrt{10 - 2\sqrt{5}}\right)$
$w_{7,1}$	$4\left(131931 + 13283\sqrt{5} - 9856\sqrt{50 - 10\sqrt{5}} + 28544\sqrt{10 - 2\sqrt{5}} + 38400\sqrt{2\left(5 + \sqrt{5}\right)}\right)$
$w_{7,2}$	$2\left(545487 + 53463\sqrt{5} + 20619\sqrt{50 - 10\sqrt{5}} + 148694\sqrt{10 - 2\sqrt{5}} - 4125\sqrt{2\left(5 + \sqrt{5}\right)} + 4950\sqrt{10\left(5 + \sqrt{5}\right)}\right)$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{7,3}$	$2\left(530397 + 58771\sqrt{5} + 19712\sqrt{50-10\sqrt{5}} + 173312\sqrt{10-2\sqrt{5}} + 39424\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{7,4}$	$650724 + 238904\sqrt{5} + 97558\sqrt{50-10\sqrt{5}} + 240212\sqrt{10-2\sqrt{5}} + 8250\sqrt{2\left(5+\sqrt{5}\right)} + 9900\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{7,5}$	$6\left(178759 - 18715\sqrt{5} + 57984\sqrt{10-2\sqrt{5}}\right)$
$w_{7,6}$	$-2\left(-453362 + 121188\sqrt{5} + 20619\sqrt{50-10\sqrt{5}} - 148694\sqrt{10-2\sqrt{5}} + 4125\sqrt{2\left(5+\sqrt{5}\right)} + 4950\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{7,7}$	$154\left(201 + 75\sqrt{5} + 256\sqrt{50-10\sqrt{5}} + 256\sqrt{10-2\sqrt{5}} - 512\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{7,8}$	$834974 - 374354\sqrt{5} - 97558\sqrt{50-10\sqrt{5}} + 240212\sqrt{10-2\sqrt{5}} + 8250\sqrt{2\left(5+\sqrt{5}\right)} - 9900\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{7,9}$	$-4\left(-235331 + 88271\sqrt{5} + 9856\sqrt{50-10\sqrt{5}} - 67968\sqrt{10-2\sqrt{5}} + 38400\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{7,10}$	$7\left(61 - 25\sqrt{5} + 32\sqrt{10-2\sqrt{5}}\right)$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
d_8	$\frac{1}{1419264}$
$w_{8,0}$	$9030 + 1778\sqrt{5}$
$w_{8,1}$	$20860 + 3848\sqrt{5} - 1232\sqrt{50 - 10\sqrt{5}} - 3850\sqrt{10 - 2\sqrt{5}} + 7680\sqrt{2\left(5 + \sqrt{5}\right)} + 1866\sqrt{10\left(5 + \sqrt{5}\right)}$
$w_{8,2}$	$53430 + 23786\sqrt{5}$
$w_{8,3}$	$2\left(20825 + 6929\sqrt{5} + 933\sqrt{50 - 10\sqrt{5}} + 3840\sqrt{10 - 2\sqrt{5}} + 1925\sqrt{2\left(5 + \sqrt{5}\right)} + 616\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{8,4}$	$16\left(3715 + 1658\sqrt{5}\right)$
$w_{8,5}$	$6\left(7455 + 3373\sqrt{5}\right)$
$w_{8,6}$	$43160 + 4496\sqrt{5}$
$w_{8,7}$	$2\left(20825 + 6929\sqrt{5} - 933\sqrt{50 - 10\sqrt{5}} - 3840\sqrt{10 - 2\sqrt{5}} - 1925\sqrt{2\left(5 + \sqrt{5}\right)} - 616\sqrt{10\left(5 + \sqrt{5}\right)}\right)$
$w_{8,8}$	$-462\left(-5 + \sqrt{5}\right)$
$w_{8,9}$	$20860 + 3848\sqrt{5} + 1232\sqrt{50 - 10\sqrt{5}} + 3850\sqrt{10 - 2\sqrt{5}} - 7680\sqrt{2\left(5 + \sqrt{5}\right)} - 1866\sqrt{10\left(5 + \sqrt{5}\right)}$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,10}$	$-7\left(-5+\sqrt{5}\right)$
d_9	$\frac{1}{35481600}$
$w_{9,0}$	$14\left(12861+25\sqrt{5}+3168\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{9,1}$	$-2\left(-530397+58771\sqrt{5}+39424\sqrt{10-2\sqrt{5}} - 173312\sqrt{2\left(5+\sqrt{5}\right)}+19712\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{9,2}$	$2\left(453362+121188\sqrt{5}-4950\sqrt{50-10\sqrt{5}}+4125\sqrt{10-2\sqrt{5}} + 148694\sqrt{2\left(5+\sqrt{5}\right)}+20619\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{9,3}$	$4\left(235331+88271\sqrt{5}+38400\sqrt{10-2\sqrt{5}} + 67968\sqrt{2\left(5+\sqrt{5}\right)}+9856\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{9,4}$	$834974+374354\sqrt{5}-9900\sqrt{50-10\sqrt{5}}-8250\sqrt{10-2\sqrt{5}} + 240212\sqrt{2\left(5+\sqrt{5}\right)}+97558\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,5}$	$6\left(178759+18715\sqrt{5}+57984\sqrt{2\left(5+\sqrt{5}\right)}\right)$
$w_{9,6}$	$1090974-106926\sqrt{5}+9900\sqrt{50-10\sqrt{5}}+8250\sqrt{10-2\sqrt{5}} + 297388\sqrt{2\left(5+\sqrt{5}\right)}-41238\sqrt{10\left(5+\sqrt{5}\right)}$

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{9,7}$	$4\left(131931 - 13283\sqrt{5} - 38400\sqrt{10-2\sqrt{5}} + 28544\sqrt{2\left(5+\sqrt{5}\right)} + 9856\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{9,8}$	$650724 - 238904\sqrt{5} + 9900\sqrt{50-10\sqrt{5}} - 8250\sqrt{10-2\sqrt{5}} + 240212\sqrt{2\left(5+\sqrt{5}\right)} - 97558\sqrt{10\left(5+\sqrt{5}\right)}$
$w_{9,9}$	$-154\left(-201 + 75\sqrt{5} - 512\sqrt{10-2\sqrt{5}} - 256\sqrt{2\left(5+\sqrt{5}\right)} + 256\sqrt{10\left(5+\sqrt{5}\right)}\right)$
$w_{9,10}$	$7\left(61 + 25\sqrt{5} + 32\sqrt{2\left(5+\sqrt{5}\right)}\right)$
d_{10}	$\frac{1}{17325}$
$w_{10,0}$	175
$w_{10,1}$	$754 - 154\sqrt{5} - 77\sqrt{10-2\sqrt{5}} + 150\sqrt{2\left(5+\sqrt{5}\right)}$
$w_{10,2}$	$25\left(47 + 5\sqrt{5}\right)$
$w_{10,3}$	$754 + 154\sqrt{5} + 150\sqrt{10-2\sqrt{5}} + 77\sqrt{2\left(5+\sqrt{5}\right)}$
$w_{10,4}$	$925 + 345\sqrt{5}$
$w_{10,5}$	1359

(*continues*)

Table C.5: Integral approximation I_j, $n = 10$. (*continued*)

d_j, w_{jk}	Expression
$w_{10,6}$	$25\left(47 - 5\sqrt{5}\right)$
$w_{10,7}$	$754 + 154\sqrt{5} - 150\sqrt{10 - 2\sqrt{5}} - 77\sqrt{2\left(5 + \sqrt{5}\right)}$
$w_{10,8}$	$925 - 345\sqrt{5}$
$w_{10,9}$	$754 - 154\sqrt{5} + 77\sqrt{10 - 2\sqrt{5}} - 150\sqrt{2\left(5 + \sqrt{5}\right)}$
$w_{10,10}$	0

Table C.6: Integral approximation I_j for $n = 12$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{276756480}$
$w_{1,0}$	$-48\left(-20198 + 5005\sqrt{2} - 54\sqrt{3} + 5005\sqrt{6}\right)$
$w_{1,1}$	$-13728\left(-17 + 9\sqrt{3}\right)$
$w_{1,2}$	$11\left(378646 + 61123\sqrt{2} - 194832\sqrt{3} - 52111\sqrt{6}\right)$
$w_{1,3}$	$4\left(1120088 + 9728\sqrt{2} - 9447\sqrt{3} - 456192\sqrt{6}\right)$
$w_{1,4}$	$-11\left(-378646 + 267971\sqrt{2} - 218864\sqrt{3} + 154737\sqrt{6}\right)$
$w_{1,5}$	$16\left(244901 + 99872\sqrt{2} - 81653\sqrt{3} - 99872\sqrt{6}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,6}$	$9667086 - 2626719\sqrt{2} + 1064016\sqrt{3} - 3182703\sqrt{6}$
$w_{1,7}$	$3899462 - 2756608\sqrt{2} + 1317506\sqrt{3} - 931840\sqrt{6}$
$w_{1,8}$	$-32\left(-172555 + 70808\sqrt{2} - 58321\sqrt{3} + 70808\sqrt{6}\right)$
$w_{1,9}$	$2\left(3377373 - 1397760\sqrt{2} + 1140637\sqrt{3} - 1378304\sqrt{6}\right)$
$w_{1,10}$	$1815054 - 1299297\sqrt{2} + 1064016\sqrt{3} - 743313\sqrt{6}$
$w_{1,11}$	$-16\left(-424107 + 299616\sqrt{2} - 141025\sqrt{3} + 99872\sqrt{6}\right)$
$w_{1,12}$	$-24\left(-38 + 35\sqrt{2} - 54\sqrt{3} + 35\sqrt{6}\right)$
d_2	$\frac{1}{276756480}$
$w_{2,0}$	$864\left(1123 - 560\sqrt{3}\right)$
$w_{2,1}$	$7678608 - 1520593\sqrt{2} - 4304518\sqrt{3} + 970017\sqrt{6}$
$w_{2,2}$	$44\left(179199 - 103424\sqrt{3}\right)$
$w_{2,3}$	$7678608 + 1520593\sqrt{2} - 4304518\sqrt{3} - 970017\sqrt{6}$
$w_{2,4}$	123552
$w_{2,5}$	$32\left(172545 + 99872\sqrt{2} - 99872\sqrt{3} - 57801\sqrt{6}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,6}$	$10733958 - 6190182\sqrt{3}$
$w_{2,7}$	$2087568 - 1481681\sqrt{2} - 1208698\sqrt{3} + 854751\sqrt{6}$
$w_{2,8}$	$32\left(245457 - 141616\sqrt{3}\right)$
$w_{2,9}$	$2087568 + 1481681\sqrt{2} - 1208698\sqrt{3} - 854751\sqrt{6}$
$w_{2,10}$	$54\left(53369 - 30775\sqrt{3}\right)$
$w_{2,11}$	$-32\left(-172545 + 99872\sqrt{2} + 99872\sqrt{3} - 57801\sqrt{6}\right)$
$w_{2,12}$	1296
d_3	$\frac{1}{17297280}$
$w_{3,0}$	$60594 - 30030\sqrt{2}$
$w_{3,1}$	$282380 - 173504\sqrt{2} - 141850\sqrt{3} + 115264\sqrt{6}$
$w_{3,2}$	$-22\left(-11177 + 6464\sqrt{2} + 6464\sqrt{3} - 3736\sqrt{6}\right)$
$w_{3,3}$	$242484 - 171072\sqrt{2} - 1930\sqrt{3} + 1216\sqrt{6}$
$w_{3,4}$	$-22\left(-11177 + 6464\sqrt{2} - 6464\sqrt{3} + 3736\sqrt{6}\right)$
$w_{3,5}$	14586

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,6}$	$591162 - 419556\sqrt{2}$
$w_{3,7}$	$242484 - 171072\sqrt{2} + 1930\sqrt{3} - 1216\sqrt{6}$
$w_{3,8}$	$400022 - 283232\sqrt{2}$
$w_{3,9}$	$282380 - 173504\sqrt{2} + 141850\sqrt{3} - 115264\sqrt{6}$
$w_{3,10}$	$100410 - 71196\sqrt{2}$
$w_{3,11}$	$565190 - 399488\sqrt{2}$
$w_{3,12}$	$57 - 105\sqrt{2}$
d_4	$\frac{1}{276756480}$
$w_{4,0}$	491616
$w_{4,1}$	$509296 + 383665\sqrt{2} + 315770\sqrt{3} + 212949\sqrt{6}$
$w_{4,2}$	$-418\left(-4013 + 2305\sqrt{3}\right)$
$w_{4,3}$	$509296 - 383665\sqrt{2} + 315770\sqrt{3} - 212949\sqrt{6}$
$w_{4,4}$	$418\left(4013 + 2305\sqrt{3}\right)$
$w_{4,5}$	$32\left(40583 + 28915\sqrt{2}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,6}$	370656
$w_{4,7}$	$509296 - 383665\sqrt{2} - 315770\sqrt{3} + 212949\sqrt{6}$
$w_{4,8}$	26720
$w_{4,9}$	$509296 + 383665\sqrt{2} - 315770\sqrt{3} - 212949\sqrt{6}$
$w_{4,10}$	12204
$w_{4,11}$	$1298656 - 925280\sqrt{2}$
$w_{4,12}$	3888
d_5	$\frac{1}{276756480}$
$w_{5,0}$	$48\left(20198 + 5005\sqrt{2} - 54\sqrt{3} - 5005\sqrt{6}\right)$
$w_{5,1}$	$2\left(1949731 + 1378304\sqrt{2} - 658753\sqrt{3} - 465920\sqrt{6}\right)$
$w_{5,2}$	$11\left(378646 + 267971\sqrt{2} - 218864\sqrt{3} - 154737\sqrt{6}\right)$
$w_{5,3}$	$2\left(3377373 + 1397760\sqrt{2} - 1140637\sqrt{3} - 1378304\sqrt{6}\right)$
$w_{5,4}$	$-11\left(-378646 + 61123\sqrt{2} - 194832\sqrt{3} + 52111\sqrt{6}\right)$
$w_{5,5}$	$16\left(424107 + 299616\sqrt{2} - 141025\sqrt{3} - 99872\sqrt{6}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{5,6}$	$3\left(3222362 + 875573\sqrt{2} - 354672\sqrt{3} - 1060901\sqrt{6}\right)$
$w_{5,7}$	$13728\left(17 + 9\sqrt{3}\right)$
$w_{5,8}$	$32\left(172555 + 70808\sqrt{2} - 58321\sqrt{3} - 70808\sqrt{6}\right)$
$w_{5,9}$	$4\left(1120088 - 9728\sqrt{2} + 9447\sqrt{3} - 456192\sqrt{6}\right)$
$w_{5,10}$	$1815054 + 1299297\sqrt{2} - 1064016\sqrt{3} - 743313\sqrt{6}$
$w_{5,11}$	$-16\left(-244901 + 99872\sqrt{2} - 81653\sqrt{3} + 99872\sqrt{6}\right)$
$w_{5,12}$	$24\left(38 + 35\sqrt{2} - 54\sqrt{3} - 35\sqrt{6}\right)$
d_6	$\frac{1}{4324320}$
$w_{6,0}$	15282
$w_{6,1}$	$16\left(2145 - 19\sqrt{2} + 891\sqrt{6}\right)$
$w_{6,2}$	$-176\left(-351 + 202\sqrt{3}\right)$
$w_{6,3}$	$16\left(2145 + 19\sqrt{2} - 891\sqrt{6}\right)$
$w_{6,4}$	$176\left(351 + 202\sqrt{3}\right)$
$w_{6,5}$	$4\left(17589 + 12484\sqrt{2}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,6}$	123120
$w_{6,7}$	$16\left(2145+19\sqrt{2}+891\sqrt{6}\right)$
$w_{6,8}$	7722
$w_{6,9}$	$-16\left(-2145+19\sqrt{2}+891\sqrt{6}\right)$
$w_{6,10}$	432
$w_{6,11}$	$70356-49936\sqrt{2}$
$w_{6,12}$	81
d_7	$\frac{1}{276756480}$
$w_{7,0}$	$-48\left(-20198+5005\sqrt{2}+54\sqrt{3}-5005\sqrt{6}\right)$
$w_{7,1}$	$2\left(3377373-1397760\sqrt{2}-1140637\sqrt{3}+1378304\sqrt{6}\right)$
$w_{7,2}$	$-11\left(-378646+267971\sqrt{2}+218864\sqrt{3}-154737\sqrt{6}\right)$
$w_{7,3}$	$3899462-2756608\sqrt{2}-1317506\sqrt{3}+931840\sqrt{6}$
$w_{7,4}$	$11\left(378646+61123\sqrt{2}+194832\sqrt{3}+52111\sqrt{6}\right)$
$w_{7,5}$	$16\left(244901+99872\sqrt{2}+81653\sqrt{3}+99872\sqrt{6}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{7,6}$	$9667086 - 2626719\sqrt{2} - 1064016\sqrt{3} + 3182703\sqrt{6}$
$w_{7,7}$	$4\left(1120088 + 9728\sqrt{2} + 9447\sqrt{3} + 456192\sqrt{6}\right)$
$w_{7,8}$	$-32\left(-172555 + 70808\sqrt{2} + 58321\sqrt{3} - 70808\sqrt{6}\right)$
$w_{7,9}$	$13728\left(17 + 9\sqrt{3}\right)$
$w_{7,10}$	$1815054 - 1299297\sqrt{2} - 1064016\sqrt{3} + 743313\sqrt{6}$
$w_{7,11}$	$-16\left(-424107 + 299616\sqrt{2} + 141025\sqrt{3} - 99872\sqrt{6}\right)$
$w_{7,12}$	$-24\left(-38 + 35\sqrt{2} + 54\sqrt{3} - 35\sqrt{6}\right)$
d_8	$\frac{1}{10250240}$
$w_{8,0}$	54048
$w_{8,1}$	$-9\left(-24784 + 1739\sqrt{2} + 6370\sqrt{3} - 6633\sqrt{6}\right)$
$w_{8,2}$	$230670 - 132858\sqrt{3}$
$w_{8,3}$	$9\left(24784 + 1739\sqrt{2} - 6370\sqrt{3} - 6633\sqrt{6}\right)$
$w_{8,4}$	$198\left(1165 + 671\sqrt{3}\right)$
$w_{8,5}$	$288\left(989 + 703\sqrt{2}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,6}$	582084
$w_{8,7}$	$9\left(24784 + 1739\sqrt{2} + 6370\sqrt{3} + 6633\sqrt{6}\right)$
$w_{8,8}$	336672
$w_{8,9}$	$-9\left(-24784 + 1739\sqrt{2} - 6370\sqrt{3} + 6633\sqrt{6}\right)$
$w_{8,10}$	13728
$w_{8,11}$	$-288\left(-989 + 703\sqrt{2}\right)$
$w_{8,12}$	144
d_9	$\frac{1}{17297280}$
$w_{9,0}$	$60594 + 30030\sqrt{2}$
$w_{9,1}$	$242484 + 171072\sqrt{2} - 1930\sqrt{3} - 1216\sqrt{6}$
$w_{9,2}$	$22\left(11177 + 6464\sqrt{2} - 6464\sqrt{3} - 3736\sqrt{6}\right)$
$w_{9,3}$	$282380 + 173504\sqrt{2} - 141850\sqrt{3} - 115264\sqrt{6}$
$w_{9,4}$	$22\left(11177 + 6464\sqrt{2} + 6464\sqrt{3} + 3736\sqrt{6}\right)$
$w_{9,5}$	$565190 + 399488\sqrt{2}$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{9,6}$	$6\left(98527 + 69926\sqrt{2}\right)$
$w_{9,7}$	$282380 + 173504\sqrt{2} + 141850\sqrt{3} + 115264\sqrt{6}$
$w_{9,8}$	$2\left(200011 + 141616\sqrt{2}\right)$
$w_{9,9}$	$242484 + 171072\sqrt{2} + 1930\sqrt{3} + 1216\sqrt{6}$
$w_{9,10}$	$6\left(16735 + 11866\sqrt{2}\right)$
$w_{9,11}$	14586
$w_{9,12}$	$57 + 105\sqrt{2}$
d_{10}	$\frac{1}{276756480}$
$w_{10,0}$	$864\left(1123 + 560\sqrt{3}\right)$
$w_{10,1}$	$2087568 + 1481681\sqrt{2} + 1208698\sqrt{3} + 854751\sqrt{6}$
$w_{10,2}$	123552
$w_{10,3}$	$2087568 - 1481681\sqrt{2} + 1208698\sqrt{3} - 854751\sqrt{6}$
$w_{10,4}$	$44\left(179199 + 103424\sqrt{3}\right)$
$w_{10,5}$	$32\left(172545 + 99872\sqrt{2} + 99872\sqrt{3} + 57801\sqrt{6}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{10,6}$	$162\left(66259 + 38211\sqrt{3}\right)$
$w_{10,7}$	$7678608 + 1520593\sqrt{2} + 4304518\sqrt{3} + 970017\sqrt{6}$
$w_{10,8}$	$32\left(245457 + 141616\sqrt{3}\right)$
$w_{10,9}$	$7678608 - 1520593\sqrt{2} + 4304518\sqrt{3} - 970017\sqrt{6}$
$w_{10,10}$	$54\left(53369 + 30775\sqrt{3}\right)$
$w_{10,11}$	$-32\left(-172545 + 99872\sqrt{2} - 99872\sqrt{3} + 57801\sqrt{6}\right)$
$w_{10,12}$	1296
d_{11}	$\frac{1}{276756480}$
$w_{11,0}$	$48\left(20198 + 5005\sqrt{2} + 54\sqrt{3} + 5005\sqrt{6}\right)$
$w_{11,1}$	$4\left(1120088 - 9728\sqrt{2} - 9447\sqrt{3} + 456192\sqrt{6}\right)$
$w_{11,2}$	$-11\left(-378646 + 61123\sqrt{2} + 194832\sqrt{3} - 52111\sqrt{6}\right)$
$w_{11,3}$	$-13728\left(-17 + 9\sqrt{3}\right)$
$w_{11,4}$	$11\left(378646 + 267971\sqrt{2} + 218864\sqrt{3} + 154737\sqrt{6}\right)$
$w_{11,5}$	$16\left(424107 + 299616\sqrt{2} + 141025\sqrt{3} + 99872\sqrt{6}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{11,6}$	$3\left(3222362 + 875573\sqrt{2} + 354672\sqrt{3} + 1060901\sqrt{6}\right)$
$w_{11,7}$	$2\left(3377373 + 1397760\sqrt{2} + 1140637\sqrt{3} + 1378304\sqrt{6}\right)$
$w_{11,8}$	$32\left(172555 + 70808\sqrt{2} + 58321\sqrt{3} + 70808\sqrt{6}\right)$
$w_{11,9}$	$2\left(1949731 + 1378304\sqrt{2} + 658753\sqrt{3} + 465920\sqrt{6}\right)$
$w_{11,10}$	$1815054 + 1299297\sqrt{2} + 1064016\sqrt{3} + 743313\sqrt{6}$
$w_{11,11}$	$-16\left(-244901 + 99872\sqrt{2} + 81653\sqrt{3} - 99872\sqrt{6}\right)$
$w_{11,12}$	$24\left(38 + 35\sqrt{2} + 54\sqrt{3} + 35\sqrt{6}\right)$
d_{12}	$\dfrac{1}{135135}$
$w_{12,0}$	945
$w_{12,1}$	$5384 - 19\sqrt{2} - 1820\sqrt{3} + 891\sqrt{6}$
$w_{12,2}$	$-2222\left(-2 + \sqrt{3}\right)$
$w_{12,3}$	$5384 + 19\sqrt{2} - 1820\sqrt{3} - 891\sqrt{6}$
$w_{12,4}$	$2222\left(2 + \sqrt{3}\right)$
$w_{12,5}$	$3121\left(2 + \sqrt{2}\right)$

(*continues*)

Table C.6: Integral approximation I_j, $n = 12$. (*continued*)

d_j, w_{jk}	Expression
$w_{12,6}$	11502
$w_{12,7}$	$5384 + 19\sqrt{2} + 1820\sqrt{3} + 891\sqrt{6}$
$w_{12,8}$	8851
$w_{12,9}$	$5384 - 19\sqrt{2} + 1820\sqrt{3} - 891\sqrt{6}$
$w_{12,10}$	3834
$w_{12,11}$	$-3121\left(-2+\sqrt{2}\right)$
$w_{12,12}$	0

Table C.7: Integral approximation I_j for $n = 14$.

d_j, w_{jk}	Expression
d_1	$\frac{1}{322882560}$
$w_{1,0}$	$66\,(12561 + 49\,c_5 - 12480\,c_6)$
$w_{1,1}$	$-2\,\big(188188\,{c_1}^2 + 156156\,{c_2}^2 + 11\,(-197253 + 896\,c_4\,(112 + 39\,c_4) + 27603\,c_5)$ $+940800\,c_6 - 572\,c_1\,(-147 + 273\,c_3 + 128\,c_6) - 104\,c_2\,(-1920 + 2128\,c_4 + 1617\,c_6)$ $+ 4\,\big(55328\,{c_3}^2 + 245280\,{c_5}^2 - 3003\,c_3\,(-35 + 32\,c_5)$ $- 56\,(1456\,c_3 + 7383\,c_4 + 8400\,c_5)\,c_6 + 551694\,{c_6}^2\big)\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,2}$	$-2\,(-49\,(52\,c_3\,(22\,c_3 - 3\,(283 + 30\,c_4)) + 55\,(847 + 19\,c_5) + 52\,(2\,(32\,c_3 - 66\,c_4 - 45\,c_5)\,c_5 + 11\,c_2\,(-1 + 2\,c_5))) - 8\,(296725\,c_3 + 49\,(-7623 + 2647\,c_5))\,c_6 + 4\,c_1\,(-195186 + 21021\,c_2 + 33761\,c_3 - 26754\,c_4 - 228193\,c_5 + 412482\,c_6))$
$w_{1,3}$	$2\,\big(88088\,{c_2}^2 - 7\,\big(12584\,{c_3}^2 - 52\,c_3\,(231 + 608\,c_5) + 3\,(-103323 + 104\,c_4\,(896 + 115\,c_4) + c_5\,(-54703 + 11960\,c_5))\big) - 308\,c_2\,(3584 + 1248\,c_4 - 4047\,c_6) - 32\,(96792 + 2288\,c_3 - 73125\,c_4 + 10192\,c_5)\,c_6 + 168168\,{c_6}^2 + 4\,c_1\,(-257994 + 80353\,c_3 - 262297\,c_5 + 470400\,c_6)\big)$
$w_{1,4}$	$-2\,\big(-2026409 + 28028\,{c_1}^2 + 1479624\,c_3 + 28028\,c_2\,(-1 + 3\,c_3) + 897603\,c_5 + 392\,(-273\,c_3\,(c_3 + c_4) + (2543\,c_3 + 585\,c_4)\,c_5) + 2747976\,c_6 - 8\,(284319\,c_3 + 232661\,c_5)\,c_6 + 28\,c_1\,(-62249 + 2002\,c_2 - 5824\,c_3 - 12012\,c_4 + 7189\,c_5 + 68674\,c_6)\big)$
$w_{1,5}$	$-2\,\big(188188\,{c_1}^2 - 156156\,{c_2}^2 - 11\,(197253 + 896\,c_4\,(112 + 39\,c_4) - 27603\,c_5) + 4854528\,c_6 - 572\,c_1\,(-147 + 273\,c_3 + 128\,c_6) + 104\,c_2\,(-1920 + 2128\,c_4 + 1617\,c_6) + 4\,\big(55328\,{c_3}^2 + 245280\,{c_5}^2 - 3003\,c_3\,(-35 + 32\,c_5) - 56\,(1456\,c_3 - 7383\,c_4 + 8400\,c_5)\,c_6 - 551694\,{c_6}^2\big)\big)$
$w_{1,6}$	$-2\,\big(28028\,c_2 + 780744\,c_3 + 196\,\big(143\,{c_1}^2 - 39\,c_3\,(11\,c_2 + 14\,(c_3 - c_4)) + 2\,(2543\,c_3 - 585\,c_4)\,c_5 + c_1\,(3377 - 286\,c_2 - 832\,c_3 + 1716\,c_4 + 1027\,c_5)\big) - 8\,(129703\,c_1 + 206241\,c_3 - 296725\,c_5)\,c_6 - 49\,(58701\,c_5 - 847\,(-55 + 72\,c_6))\big)$
$w_{1,7}$	$2\,\big(-88088\,{c_2}^2 + 21\,(103323 + 104\,c_4\,(896 + 115\,c_4) + 54703\,c_5) + 308\,c_2\,(3584 + 1248\,c_4 - 4047\,c_6) - 2697984\,c_6 + 4\,c_1\,(-257994 + 80353\,c_3 - 262297\,c_5 + 470400\,c_6) - 52\,\big(7\,\big(11\,c_3\,(-21 + 22\,c_3) - 608\,c_3\,c_5 + 690\,{c_5}^2\big) + 8\,(176\,c_3 + 5625\,c_4 + 784\,c_5)\,c_6 + 3234\,{c_6}^2\big)\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{1,8}$	$2\,(7\,(52\,c_3\,(154\,c_3+15\,(295+42\,c_4))+11\,(26317+31897\,c_5) - 364\,(11\,c_2\,(-1+2\,c_5)+2\,c_5\,(-32\,c_3-66\,c_4+45\,c_5))) + 4\,c_1\,(369906+21021\,c_2-33761\,c_3-26754\,c_4+228193\,c_5-568638\,c_6) - 8\,(343497+232661\,c_3+240359\,c_5)\,c_6)$
$w_{1,9}$	$-2\,\left(68068\,{c_1}^2-1031976\,c_3+312\,(640-427\,c_4)\,c_4 - 1092\,c_2\,(1792+87\,c_4)-77\,(28179+8367\,c_5-23296\,c_6) - 364\,c_1\,(1155+608\,c_3-1207\,c_5-896\,c_6)+8\,\left(427\,c_3\,(39\,c_3-313\,c_5) + (286858\,c_2+235200\,c_3-143\,(483\,c_4+64\,c_5))\,c_6+175329\,{c_6}^2\right)\right)$
$w_{1,10}$	$2\,\left(28028\,{c_1}^2+1493841\,c_5-28028\,c_2\,(1+2\,c_3+3\,c_5)-196\,\left(832\,{c_3}^2 - c_3\,(3377+1716\,c_4+624\,c_5-5294\,c_6)-6\,c_5\,(91\,c_4+800\,c_5-1403\,c_6)\right) - 41503\,(-55+72\,c_6)-52\,c_1\,(539\,c_3-147\,(283+30\,c_4)+45650\,c_6)\right)$
$w_{1,11}$	$2\,(2865423+256256\,c_1-442624\,c_3+1960623\,c_5-4526336\,c_6)$
$w_{1,12}$	$2\,\left(2026409+28028\,{c_1}^2+2192721\,c_5+28028\,c_2\,(1+2\,c_3+3\,c_5) - 52\,c_1\,(30975+539\,c_3+4410\,c_4-35794\,c_6)-2747976\,c_6-28\,\left(5824\,{c_3}^2 + c_3\,(62249+12012\,c_4-4368\,c_5-68674\,c_6)+6\,c_5\,(637\,c_4-5600\,c_5+13539\,c_6)\right)\right)$
$w_{1,13}$	$-2\,\left(68068\,{c_1}^2+312\,c_4\,(-640+427\,c_4)-77\,(28179+8367\,c_5-51968\,c_6) + 52\,c_2\,(37632+1827\,c_4-44132\,c_6)-364\,c_1\,(1155+608\,c_3-1207\,c_5-896\,c_6) + 8\,\left(16653\,{c_3}^2+11\,(6279\,c_4-832\,c_5-15939\,c_6)\,c_6 + c_3\,(-128997-133651\,c_5+235200\,c_6)\right)\right)$
$w_{1,14}$	$33\,(17+49\,c_5-64\,c_6)$
d_2	$\dfrac{1}{322882560}$
$w_{2,0}$	$-3234\,(-257+c_3+256\,c_5)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,1}$	$-2\left(56056\,c_1{}^2 - 156\,c_2\,(-1335 + 539\,c_3 + 1078\,c_5) - 52\,c_1\,(-1617 + 2058\,c_2 + 3136\,c_3 - 6468\,c_4 + 219\,c_5 + 4410\,c_6) - 49\left(4680\,c_3{}^2 + c_3\,(5973 - 1144\,c_4 - 13116\,c_5) + 88\,c_4\,(-256 + 321\,c_5) + 3\,(14861 + 8\,c_5\,(-3355 + 1600\,c_5 - 1859\,c_6) + 13312\,c_6)\right)\right)$
$w_{2,2}$	$14\left(315403 + 38064\,c_1{}^2 + 4\,c_1\,(25173 + 28501\,c_3 - 80558\,c_5) + 19\,c_3\,(-10277 + 15964\,c_5) + 4\,c_5\,(-130361 + 40149\,c_5)\right)$
$w_{2,3}$	$-2\left(107016\,c_1{}^2 + 4312\,c_2\,(256 + 13\,c_3 - 321\,c_5) + 392\,c_1\,(2673 + 858\,c_2 + 2257\,c_3 + 585\,c_4 - 4384\,c_5 - 273\,c_6) + 3\,(49\,(-14861 + 13312\,c_4 + 16852\,c_5) + 2548\,c_5\,(-286\,c_4 + 39\,c_5 - 22\,c_6) + 69420\,c_6 + c_3\,(115908\,c_5 - 539\,(493 + 52\,c_6)))\right)$
$w_{2,4}$	$14\left(295955 + 248996\,c_1 + 23296\,c_1{}^2 - 123367\,c_3 - 23296\,c_1\,c_3 - 117852\,c_3{}^2 - 4\,(155667 + 72727\,c_1 - 81234\,c_3)\,c_5 + 259116\,c_5{}^2\right)$
$w_{2,5}$	$-2\left(56056\,c_1{}^2 + 156\,c_2\,(-1335 + 539\,c_3 + 1078\,c_5) + 52\,c_1\,(1617 + 2058\,c_2 - 3136\,c_3 - 6468\,c_4 - 219\,c_5 + 4410\,c_6) - 49\left(4680\,c_3{}^2 + c_3\,(5973 + 1144\,c_4 - 13116\,c_5) - 88\,c_4\,(-256 + 321\,c_5) + 3\,(14861 - 13312\,c_6 + 8\,c_5\,(-3355 + 1600\,c_5 + 1859\,c_6))\right)\right)$
$w_{2,6}$	$-14\left(-315403 + 86528\,c_1{}^2 - 13081\,c_3 + 162468\,c_3{}^2 + 4\,c_1\,(23639 + 21632\,c_3 - 52265\,c_5) - 117852\,(-1 + 2\,c_3)\,c_5 + 351780\,c_5{}^2\right)$
$w_{2,7}$	$-2\left(107016\,c_1{}^2 - 147\,(14861 + 5423\,c_3 + 13312\,c_4 - 16852\,c_5) - 392\,c_1\,(-2673 + 858\,c_2 - 2257\,c_3 + 585\,c_4 + 4384\,c_5 - 273\,c_6) - 208260\,c_6 + 4\,(-1078\,c_2\,(256 + 13\,c_3 - 321\,c_5) + 39\,c_5\,(2229\,c_3 + 14014\,c_4 + 1911\,c_5) + 21021\,(c_3 + 2\,c_5)\,c_6)\right)$
$w_{2,8}$	$14\left(295955 + 217692\,c_1 + 324425\,c_3 + 141148\,c_1\,c_3 - 4\,(35893 + 75410\,c_1 + 70603\,c_3)\,c_5 - 267612\,c_5{}^2\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{2,9}$	$-2\left(163072\,{c_1}^2+3\left(-49\,(14861+13312\,c_2+c_3\,(11979+6400\,c_3))\right.\right.$ $-52\,(-1335+539\,c_3)\,c_4+196\,(5071+3718\,c_2+3200\,c_3-286\,c_4)\,c_5-39468\,{c_5}^2\big)$ $-4312\,(256+13\,c_3-321\,c_5)\,c_6$ $-2548\,c_1\,(165+90\,c_2-48\,c_3+42\,c_4-139\,c_5+132\,c_6)\big)$
$w_{2,10}$	$14\left(315403+48464\,{c_1}^2+48464\,{c_3}^2+52\,c_1\,(5943+1664\,c_3-6033\,c_5)\right.$ $\left.+c_3\,(208329+1872\,c_5)-12\,c_5\,(27183+19642\,c_5)\right)$
$w_{2,11}$	$98\,(59103+6656\,c_1+38433\,c_3)-9199104\,c_5$
$w_{2,12}$	$-14\left(-295955+23296\,{c_1}^2-23296\,{c_3}^2+c_3\,(154671-385464\,c_5)\right.$ $\left.+52\,c_1\,(4425+448\,c_3-4983\,c_5)+12\,c_5\,(14573+27078\,c_5)\right)$
$w_{2,13}$	$-2\left(163072\,{c_1}^2-15288\,c_2\,(-128+143\,c_5)\right.$ $+3\left(-49\,(14861+c_3\,(11979+6400\,c_3))+52\,(-1335+539\,c_3)\,c_4\right.$ $\left.+196\,(5071+3200\,c_3+286\,c_4)\,c_5-39468\,{c_5}^2\right)+4312\,(256+13\,c_3-321\,c_5)\,c_6$ $\left.+2548\,c_1\,(-165+90\,c_2+48\,c_3+42\,c_4+139\,c_5+132\,c_6)\right)$
$w_{2,14}$	$-1617\,(-1+c_3)$
d_3	$\dfrac{1}{322882560}$
$w_{3,0}$	$66\,(12561+49\,c_1-12480\,c_4)$
$w_{3,1}$	$2\left(-133224\,{c_2}^2+c_1\,(644259+439348\,c_3+73216\,c_4-1069208\,c_5)\right.$ $-11\,(7\,(-28179+52\,c_3\,(105+17\,c_3)+51968\,c_4)+93816\,c_5)$ $+56\left(25047\,{c_4}^2+33600\,c_4\,c_5-2379\,{c_5}^2+208\,c_3\,(28\,c_4+19\,c_5)\right)$ $\left.+156\,c_2\,(-1280+3542\,c_4-609\,c_6)+1956864\,c_6-2294864\,c_4\,c_6\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,2}$	$2\,\big(940800\,c_1{}^2 + 28028\,c_3{}^2 + 52\,c_3\,(4410\,c_2 + 45650\,c_4 - 49\,(849 + 11\,c_5)) - 3\,c_1\,(-497947 + 35672\,c_2 + 549976\,c_4 + 40768\,c_5 - 28028\,c_6) + 49\,(8\,c_4\,(-7623 + 2647\,c_5) + 11\,(4235 + 52\,c_6) - 4\,c_5\,(3377 - 1716\,c_2 + 832\,c_5 + 286\,c_6))\big)$
$w_{3,3}$	$-2\,\big(981120\,c_1{}^2 - 384384\,c_2{}^2 + 21\,(-103323 + 231168\,c_4 - 20020\,c_5) + c_1\,(-1881600\,c_4 + 33\,(9201 + 11648\,c_5)) + 199680\,c_6 + 52\,\big(3619\,c_3{}^2 - 42438\,c_4{}^2 + 11\,c_3\,(-147 + 128\,c_4 - 273\,c_5) + 6272\,c_4\,c_5 + 4256\,c_5{}^2 - 3234\,c_4\,c_6 - 3003\,c_6{}^2\big) - 224\,c_2\,(7383\,c_4 - 4\,(1232 + 247\,c_6))\big)$
$w_{3,4}$	$-2\,\big(229320\,c_1{}^2 + 4\,(-26\,c_5\,(2205\,c_2 + 17897\,c_4 + 539\,c_5) + c_3\,(369906 + 26754\,c_2 - 568638\,c_4 + 33761\,c_5 - 21021\,c_6)) + 7\,c_1\,(-350867 + 48048\,c_2 + 130396\,c_3 + 274696\,c_4 + 23296\,c_5 - 8008\,c_6) + 7\,(-289487 + 392568\,c_4 + 230100\,c_5 + 4004\,c_6)\big)$
$w_{3,5}$	$2\,\big(312\,c_2\,(640 + 427\,c_2) + c_1\,(644259 + 439348\,c_3 + 73216\,c_4 - 1069208\,c_5) - 11\,(-197253 + 38220\,c_3 + 163072\,c_4 + 93816\,c_5) - 28\,\big(2431\,c_3{}^2 - 416\,c_3\,(28\,c_4 + 19\,c_5) + 6\,\big(253\,c_4\,(13\,c_2 + 33\,c_4) - 11200\,c_4\,c_5 + 793\,c_5{}^2\big)\big) - 1956864\,c_6 + 52\,(1827\,c_2 + 44132\,c_4)\,c_6\big)$
$w_{3,6}$	$-2\,\big(229320\,c_1{}^2 - 49\,c_1\,(1045 + 6864\,c_2 - 18628\,c_3 + 21176\,c_4 - 3328\,c_5 - 1144\,c_6) + 49\,(-46585 + 60984\,c_4 - 44148\,c_5 - 572\,c_6) + 4\,(26\,(2205\,c_2 + 22825\,c_4 - 539\,c_5)\,c_5 + c_3\,(195186 - 26754\,c_2 - 412482\,c_4 + 33761\,c_5 + 21021\,c_6))\big)$
$w_{3,7}$	$-2\,\big(981120\,c_1{}^2 + 384384\,c_2{}^2 + 21\,(-103323 + 44800\,c_4 - 20020\,c_5) + c_1\,(-1881600\,c_4 + 33\,(9201 + 11648\,c_5)) - 199680\,c_6 + 52\,\big(3619\,c_3{}^2 + 42438\,c_4{}^2 + 11\,c_3\,(-147 + 128\,c_4 - 273\,c_5) + 6272\,c_4\,c_5 + 4256\,c_5{}^2 + 3234\,c_4\,c_6 + 3003\,c_6{}^2\big) + 224\,c_2\,(7383\,c_4 - 4\,(1232 + 247\,c_6))\big)$
$w_{3,8}$	$2\,\big(940800\,c_1{}^2 + 28028\,c_3{}^2 - 52\,c_3\,(-30975 + 4410\,c_2 + 35794\,c_4 + 539\,c_5) + 3\,c_1\,(730907 + 35672\,c_2 - 758184\,c_4 - 40768\,c_5 - 28028\,c_6) + 7\,(289487 - 8\,c_4\,(49071 + 34337\,c_5) - 4004\,c_6 + 4\,c_5\,(62249 - 12012\,c_2 - 5824\,c_5 + 2002\,c_6))\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{3,9}$	$-2\,\big(251160\,{c_1}^2 + 251160\,{c_2}^2 + 3097344\,c_4 - 7\,c_1\,(164109 + 149884\,c_3 - 46592\,c_4 - 31616\,c_5) + 77\,(-28179 + 1092\,c_5 - 14336\,c_6) + 1248\,c_2\,(-1568 + 1875\,c_4 + 308\,c_6) + 4\,\big(c_3\,(-257994 + 470400\,c_4 - 80353\,c_5) - 11\,\big(3822\,{c_4}^2 + c_4\,(1664\,c_5 - 28329\,c_6) - 2002\,(c_5 - c_6)\,(c_5 + c_6)\big)\big)\big)$
$w_{3,10}$	$2\,\big(2282665 - 2988216\,c_4 + 780744\,c_5 + c_1\,(2876349 - 229320\,c_2 + 201292\,c_3 - 2373800\,c_4 + 996856\,c_5) - 196\,\big(143\,{c_3}^2 + c_3\,(-3377 + 1716\,c_2 + 5294\,c_4 - 832\,c_5 - 286\,c_6) + 3\,c_5\,(182\,c_2 + 2806\,c_4 - 182\,c_5 - 143\,c_6)\big) + 28028\,c_6\big)$
$w_{3,11}$	$2\,(2865423 + 1960623\,c_1 - 256256\,c_3 - 4526336\,c_4 + 442624\,c_5)$
$w_{3,12}$	$2\,\big(2026409 - 2747976\,c_4 + 1479624\,c_5 + c_1\,(-897603 + 229320\,c_2 + 201292\,c_3 + 1861288\,c_4 + 996856\,c_5) + 28\,\big(-1001\,{c_3}^2 + c_3\,(-62249 + 12012\,c_2 + 68674\,c_4 + 5824\,c_5 - 2002\,c_6) + 3\,c_5\,(1274\,c_2 - 27078\,c_4 + 1274\,c_5 - 1001\,c_6)\big) - 28028\,c_6\big)$
$w_{3,13}$	$-2\,\big(251160\,{c_1}^2 - 251160\,{c_2}^2 + 2697984\,c_4 - 7\,c_1\,(164109 + 149884\,c_3 - 46592\,c_4 - 31616\,c_5) - 1248\,c_2\,(-1568 + 1875\,c_4 + 308\,c_6) + 77\,(-28179 + 1092\,c_5 + 14336\,c_6) + 4\,\big(c_3\,(-257994 + 470400\,c_4 - 80353\,c_5) + 11\,\big(3822\,{c_4}^2 - c_4\,(1664\,c_5 + 28329\,c_6) + 2002\,\big({c_5}^2 + {c_6}^2\big)\big)\big)\big)$
$w_{3,14}$	$33\,(17 + 49\,c_1 - 64\,c_4)$
d_4	$\dfrac{1}{46126080}$
$w_{4,0}$	$462\,(257 + c_1 - 256\,c_3)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,1}$	$14\,\big(44583 - 67716\,c_3 + 7228\,c_3{}^2 - 22528\,c_4 + 28248\,c_3\,c_4 + 156\,c_2\,(-25 + 22\,c_3 - 14\,c_5) - 21384\,c_5 + 41728\,c_3\,c_5 + 6864\,c_4\,c_5 - 2184\,c_5{}^2 + c_1\,(-16269 + 1716\,c_2 - 4108\,c_3 - 1144\,c_4 + 18056\,c_5) - 312\,(-128 + 143\,c_3 + 15\,c_5)\,c_6\big)$
$w_{4,2}$	$-14\,\big(-47509 + 16836\,c_1{}^2 - 47476\,c_3{}^2 + c_3\,(105132 - 23492\,c_5) + c_1\,(-3361 + 33672\,c_3 - 3328\,c_5) + 4\,(3377 - 832\,c_5)\,c_5\big)$
$w_{4,3}$	$14\,\big(19200\,c_1{}^2 - 3\,(-14861 + 19140\,c_3 + 13312\,c_4 - 2860\,c_5 + 1300\,c_6) + c_1\,(-35937 - 1144\,c_2 + 38400\,c_3 + 2496\,c_5 + 1716\,c_6) + 4\,\big(22\,c_2\,(-256 + 321\,c_3 + 78\,c_5) - 13\,\big(11\,c_3{}^2 + 2\,c_5\,(-45\,c_4 + 32\,c_5 + 21\,c_6) - 3\,c_3\,(286\,c_4 - 39\,c_5 + 22\,c_6)\big)\big)\big)$
$w_{4,4}$	$2\,\big(278795 + 48464\,c_1{}^2 + 324936\,c_3{}^2 + c_1\,(135223 - 124592\,c_3 - 86528\,c_5) + 52\,c_5\,(-4425 + 932\,c_5) + 100\,c_3\,(-5931 + 2405\,c_5)\big)$
$w_{4,5}$	$-14\,\big(-44583 + 67716\,c_3 - 7228\,c_3{}^2 - 22528\,c_4 + 28248\,c_3\,c_4 + c_1\,(16269 + 1716\,c_2 + 4108\,c_3 - 1144\,c_4 - 18056\,c_5) + 156\,c_2\,(-25 + 22\,c_3 - 14\,c_5) + 21384\,c_5 - 41728\,c_3\,c_5 + 6864\,c_4\,c_5 + 2184\,c_5{}^2 - 312\,(-128 + 143\,c_3 + 15\,c_5)\,c_6\big)$
$w_{4,6}$	$14\,\big(47509 + 3328\,c_1{}^2 + 33672\,c_3{}^2 + 52\,(849 - 64\,c_5)\,c_5 + c_1\,(-26983 + 37000\,c_3 + 3328\,c_5) - 4\,c_3\,(19455 + 11869\,c_5)\big)$
$w_{4,7}$	$14\,\big(19200\,c_1{}^2 - 88\,c_2\,(-256 + 321\,c_3 + 78\,c_5) + c_1\,(-35937 + 1144\,c_2 + 38400\,c_3 + 2496\,c_5 - 1716\,c_6) + 3\,(14861 + 13312\,c_4 + 2860\,c_5 + 1300\,c_6) - 4\,\big(143\,c_3{}^2 + 26\,c_5\,(45\,c_4 + 32\,c_5 - 21\,c_6) + 3\,c_3\,(4785 + 3718\,c_4 + 507\,c_5 + 286\,c_6)\big)\big)$
$w_{4,8}$	$-2\,\big(-278795 + 162468\,c_1{}^2 + 278564\,c_3{}^2 + c_1\,(-86759 + 324936\,c_3 - 86528\,c_5) + 4\,c_5\,(-62249 + 21632\,c_5) + 4\,c_3\,(40617 + 83881\,c_5)\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{4,9}$	$14\left(44583 + 4680\,c_1{}^2 - 37752\,c_3 - 38400\,c_3{}^2 - 3900\,c_4 + 3432\,c_3\,c_4 - 1716\,c_5 + 2756\,c_3\,c_5 - 2184\,c_4\,c_5 - 1144\,c_5{}^2 - 312\,c_2\,(-128 + 143\,c_3 + 15\,c_5) - 88\,(-256 + 321\,c_3 + 78\,c_5)\,c_6 + c_1\,(-5973 - 26428\,c_3 + 1716\,c_4 - 3328\,c_5 + 1144\,c_6)\right)$
$w_{4,10}$	$-14\,(-47509 - 16836\,c_5 + 4\,c_3\,(11869 + 5041\,c_3 + 9250\,c_5) + c_1\,(-30673 + 50804\,c_3 + 20164\,c_5))$
$w_{4,11}$	$14\,(59103 - 38433\,c_1 - 90880\,c_3 + 6656\,c_5)$
$w_{4,12}$	$2\left(278795 + 287060\,c_3{}^2 + c_1\,(-343873 + 230100\,c_3 - 114004\,c_5) + 12\,c_5\,(16711 + 3172\,c_5) - 4\,c_3\,(160391 + 59602\,c_5)\right)$
$w_{4,13}$	$14\left(44583 + 4680\,c_1{}^2 + 3900\,c_4 - 24\,c_3\,(1600\,c_3 + 143\,(11 + c_4)) - 1716\,c_5 + 52\,(53\,c_3 + 42\,c_4)\,c_5 - 1144\,c_5{}^2 + 312\,c_2\,(-128 + 143\,c_3 + 15\,c_5) - 22528\,c_6 + 264\,(107\,c_3 + 26\,c_5)\,c_6 - c_1\,(5973 + 26428\,c_3 + 1716\,c_4 + 3328\,c_5 + 1144\,c_6)\right)$
$w_{4,14}$	$231\,(1 + c_1)$
d_5	$\dfrac{1}{322882560}$
$w_{5,0}$	$-66\,(-12561 + 12480\,c_2 + 49\,c_3)$
$w_{5,1}$	$-2\left(88088\,c_1{}^2 - 168168\,c_2{}^2 - 77\,(28179 + 8\,c_4\,(-1792 + 143\,c_4)) - 52\,c_1\,(1408\,c_2 + 7\,(-231 + 608\,c_3 - 883\,c_5)) + 7\,c_3\,(164109 + 35880\,c_3 - 149884\,c_5) + 1031976\,c_5 - 4\,c_2\,(81536\,c_3 + 3\,(-258112 + 103873\,c_4 + 156800\,c_5 - 195000\,c_6)) - 34944\,(56 + 11\,c_4)\,c_6 + 251160\,c_6{}^2\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{5,2}$	$2\left(107016\,c_1{}^2 + 8\,c_2\,(296725\,c_3 + 49\,(-7623 + 2647\,c_5)) - 4\,c_1\,(-195186 + 412482\,c_2 + 249214\,c_3 + 21021\,c_4 + 40768\,c_5 + 26754\,c_6) + 49\,(11\,(4235 + 52\,c_4\,(-1 + 2\,c_5) - 4\,c_5\,(307 + 13\,c_5 - 156\,c_6)) + c_3\,(-58701 + 4108\,c_5 + 4680\,c_6))\right)$
$w_{5,3}$	$-2\left(133224\,c_1{}^2 - 8\,c_1\,(-128997 + 235200\,c_2 + 133651\,c_3 - 27664\,c_5) + 21\,(-103323 + 30679\,c_3 + 93184\,c_4 - 20020\,c_5) + 199680\,c_6 - 4\left(350658\,c_2{}^2 - 4\,c_2\,(250096 + 4576\,c_3 - 143429\,c_4 + 20384\,c_5) + 138138\,c_2\,c_6 + 91\left(1207\,c_3\,c_5 - 187\,c_5{}^2 + 261\,c_4\,c_6 - 366\,c_6{}^2\right)\right)\right)$
$w_{5,4}$	$-2\left(163072\,c_1{}^2 - 3\,c_3\,(-730907 + 313600\,c_3 - 28028\,c_4) - 8\,c_2\,(-343497 + 284319\,c_3 + 232661\,c_5) - 7\,(289487 + 4004\,c_4 + 52\,c_5\,(-4425 + 77\,c_5)) + 15288\,(7\,c_3 - 15\,c_5)\,c_6 + 28\,c_1\,(-62249 + 68674\,c_2 - 4368\,c_3 + 2002\,c_4 - 1001\,c_5 + 12012\,c_6)\right)$
$w_{5,5}$	$-2\left(88088\,c_1{}^2 + 168168\,c_2{}^2 - 52\,c_1\,(1408\,c_2 + 7\,(-231 + 608\,c_3 - 883\,c_5)) + 7\,c_3\,(164109 + 35880\,c_3 - 149884\,c_5) + 11\,(-197253 + 56\,c_4\,(-1792 + 143\,c_4) + 93816\,c_5) + 34944\,(56 + 11\,c_4)\,c_6 - 251160\,c_6{}^2 - 4\,c_2\,(-674496 + 81536\,c_3 - 311619\,c_4 + 470400\,c_5 + 585000\,c_6)\right)$
$w_{5,6}$	$2\left(-163072\,c_1{}^2 + 3\,c_3\,(-497947 + 313600\,c_3 + 28028\,c_4) + 8\,c_2\,(-373527 + 206241\,c_3 - 296725\,c_5) + 49\,(46585 - 572\,c_4 + 52\,c_5\,(849 + 11\,c_5)) + 15288\,(7\,c_3 - 15\,c_5)\,c_6 + 196\,c_1\,(-3377 + 5294\,c_2 + 624\,c_3 + 286\,c_4 + 143\,c_5 + 1716\,c_6)\right)$
$w_{5,7}$	$-2\left(133224\,c_1{}^2 - 8\,c_1\,(-128997 + 235200\,c_2 + 133651\,c_3 - 27664\,c_5) + 21\,(-103323 + 30679\,c_3 - 93184\,c_4 - 20020\,c_5) - 199680\,c_6 + 4\left(350658\,c_2{}^2 + 26\,c_2\,(17248 + 704\,c_3 + 22066\,c_4 + 3136\,c_5 + 5313\,c_6) - 91\left(1207\,c_3\,c_5 - 187\,c_5{}^2 - 261\,c_4\,c_6 + 366\,c_6{}^2\right)\right)\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{5,8}$	$2\,\big(107016\,c_1{}^2 - 8\,c_2\,(343497 + 232661\,c_3 + 240359\,c_5) - 4\,c_1\,(-369906 + 568638\,c_2 + 249214\,c_3 - 21021\,c_4 + 40768\,c_5 - 26754\,c_6) + 7\,(11\,(26317 + 364\,c_4 + 22636\,c_5) + c_3\,(128229 + 28756\,c_5 - 32760\,c_6) - 4004\,c_5\,(2\,c_4 + c_5 + 12\,c_6))\big)$
$w_{5,9}$	$-2\,\big(221312\,c_1{}^2 + 2206776\,c_2{}^2 + 3\,c_3\,(-101211 + 327040\,c_3) + 364\,c_1\,(896\,c_2 - 33\,(35 + 32\,c_3 - 13\,c_5)) + 77\,(-28179 + 52\,c_5\,(21 + 47\,c_5) - 14336\,c_6) + 8\,c_2\,(117600 + 235200\,c_3 - 21021\,c_4 - 9152\,c_5 + 206724\,c_6) + 52\,\big(3003\,c_4{}^2 + 7392\,c_6{}^2 + 32\,c_4\,(120 + 133\,c_6)\big)\big)$
$w_{5,10}$	$2\,\big(56056\,c_1{}^2 - 539\,(-4235 + 95\,c_3 + 52\,c_4) + 780744\,c_5 - 52\,c_1\,(45650\,c_2 - 49\,(849 + 64\,c_3 + 53\,c_5 - 90\,c_6)) - 196\,\big(1170\,c_3{}^2 + 286\,c_3\,c_4 + 4657\,c_3\,c_5 + 429\,c_4\,c_5 + 2\,c_2\,(7623 + 2647\,c_3 + 4209\,c_5) + 78\,(22\,c_3 + 7\,c_5)\,c_6\big)\big)$
$w_{5,11}$	$2\,(2865423 + 442624\,c_1 - 4526336\,c_2 - 1960623\,c_3 + 256256\,c_5)$
$w_{5,12}$	$2\,\big(2026409 + 56056\,c_1{}^2 - 2456069\,c_3 + 28028\,c_4 + 1479624\,c_5 + 52\,c_1\,(35794\,c_2 + 7\,(-4425 + 448\,c_3 + 371\,c_5 + 630\,c_6)) + 28\,\big(c_2\,(-98142 + 68674\,c_3 - 81234\,c_5) + 7\,\big(-1170\,c_3{}^2 + 286\,c_3\,c_4 - 4657\,c_3\,c_5 + 429\,c_4\,c_5 + 78\,(22\,c_3 + 7\,c_5)\,c_6\big)\big)\big)$
$w_{5,13}$	$-2\,\big(221312\,c_1{}^2 - 2206776\,c_2{}^2 + 3\,c_3\,(-101211 + 327040\,c_3) + 364\,c_1\,(896\,c_2 - 33\,(35 + 32\,c_3 - 13\,c_5)) + 8\,c_2\,(606816 + 235200\,c_3 + 21021\,c_4 - 9152\,c_5 - 206724\,c_6) + 77\,(-28179 + 52\,c_5\,(21 + 47\,c_5) + 14336\,c_6) - 52\,\big(3003\,c_4{}^2 + 7392\,c_6{}^2 + 32\,c_4\,(120 + 133\,c_6)\big)\big)$
$w_{5,14}$	$-33\,(-17 + 64\,c_2 + 49\,c_3)$
d_6	$\dfrac{1}{322882560}$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,0}$	$-3234\,(-257+256\,c_1-c_5)$
$w_{6,1}$	$2\,\big(118404\,c_1{}^2-156\,c_2\,(1335+686\,c_3+539\,c_5) + 196\,c_1\,(858\,c_2+1807\,c_3+3\,(-5071+2354\,c_4+3200\,c_5-3718\,c_6)) - 49\,\big(3328\,c_3{}^2-88\,c_4\,(-256+13\,c_5)-156\,c_3\,(-55+44\,c_4-16\,c_5-30\,c_6) - 3\,(14861+c_5\,(-11979+6400\,c_5)+13312\,c_6)\big)\big)$
$w_{6,2}$	$-14\,\big(235704\,c_1{}^2-156\,c_1\,(-2091+2011\,c_3-12\,c_5) + 11\,(-28673+18939\,c_5)-52\,\big(c_3\,(-5943+932\,c_3)+1664\,c_3\,c_5+932\,c_5{}^2\big)\big)$
$w_{6,3}$	$2\,\big(1881600\,c_1{}^2+49\,\big(88\,c_2\,(-256+78\,c_3+13\,c_5)-3\,(-14861+13312\,c_4+1991\,c_5) + 52\,\big(-22\,c_3{}^2+90\,c_5{}^2+c_3\,(33+90\,c_4+64\,c_5)\big)\big)-156\,(1335+686\,c_3+539\,c_5)\,c_6 + 12\,c_1\,(115346\,c_2-949\,c_3+49\,(-6710+3718\,c_4+1093\,c_5+286\,c_6))\big)$
$w_{6,4}$	$-14\,(-295955+4\,c_1\,(35893+66903\,c_1-75410\,c_3) + 217692\,c_3+324425\,c_5-4\,(70603\,c_1+35287\,c_3)\,c_5)$
$w_{6,5}$	$2\,\big(118404\,c_1{}^2+156\,c_2\,(1335+686\,c_3+539\,c_5) - 196\,c_1\,(858\,c_2-1807\,c_3+3\,(5071+2354\,c_4-3200\,c_5-3718\,c_6)) - 49\,\big(3328\,c_3{}^2+88\,c_4\,(-256+13\,c_5)+156\,c_3\,(55+44\,c_4+16\,c_5-30\,c_6) + 3\,(-14861+(11979-6400\,c_5)\,c_5+13312\,c_6)\big)\big)$
$w_{6,6}$	$14\,\big(315403+160596\,c_1{}^2+4\,c_1\,(-130361+80558\,c_3-75829\,c_5) + 195263\,c_5+4\,c_3\,(-25173+9516\,c_3+28501\,c_5)\big)$
$w_{6,7}$	$2\,\big(1103872\,c_2+147\,(14861+572\,c_3+13312\,c_4-1991\,c_5) + 208260\,c_6+4\,\big(470400\,c_1{}^2 - 637\,\big(22\,c_3{}^2+22\,c_2\,(6\,c_3+c_5)+2\,c_3\,(45\,c_4-32\,c_5-21\,c_6)-3\,c_5\,(30\,c_5+11\,c_6)\big) - 3\,c_1\,(115346\,c_2+949\,c_3+49\,(6710+3718\,c_4-1093\,c_5+286\,c_6))\big)\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{6,8}$	$-14\left(-230100\,c_3 - 11\,(26905 + 14061\,c_5) + 4\left(81234\,c_1{}^2 + 3\,c_1\,(14573 + 21593\,c_3 + 32122\,c_5) + 5824\left(c_3{}^2 + c_3\,c_5 - c_5{}^2\right)\right)\right)$
$w_{6,9}$	$-2\left(298116\,c_1{}^2 - 1617\,(1351 + 648\,c_3) + 208260\,c_4 + 15288\,(c_2\,(-128 + 15\,c_3) + 7\,c_3\,(c_3 + c_4)) + 4\,c_1\,(49\,(12639 + 11154\,c_2 + 8768\,c_3 - 858\,c_4) - 86931\,c_5) + 49\,(18056\,c_3 + 33\,(493 + 52\,c_4))\,c_5\right) - 8624\,(-256 + 321\,c_1 + 78\,c_3 + 13\,c_5)\,c_6$
$w_{6,10}$	$-14\left(-315403 + 351780\,c_1{}^2 + 86528\,c_3{}^2 + 4\,c_3\,(-23639 + 21632\,c_5) + c_5\,(13081 + 162468\,c_5) + 4\,c_1\,(52265\,c_3 + 29463\,(1 + 2\,c_5))\right)$
$w_{6,11}$	$-2\,(4599552\,c_1 + 49\,(-59103 + 6656\,c_3 + 38433\,c_5))$
$w_{6,12}$	$14\left(295955 + 259116\,c_1{}^2 + 23296\,c_3{}^2 + 4\,c_1\,(-155667 + 72727\,c_3 - 81234\,c_5) + (123367 - 117852\,c_5)\,c_5 - 4\,c_3\,(62249 + 5824\,c_5)\right)$
$w_{6,13}$	$2\left(-298116\,c_1{}^2 + 15288\,c_2\,(-128 + 15\,c_3) + 208260\,c_4 - 539\,(-4053 + 1479\,c_5 + 2048\,c_6) + 4\,c_1\,(-619311 + 546546\,c_2 - 429632\,c_3 - 42042\,c_4 + 86931\,c_5 + 346038\,c_6) + 196\left(-546\,c_3{}^2 + 143\,c_5\,(3\,c_4 + 2\,c_6) + 2\,c_3\,(2673 + 273\,c_4 - 2257\,c_5 + 858\,c_6)\right)\right)$
$w_{6,14}$	$1617\,(1 + c_5)$
d_7	$\dfrac{1}{10090080}$
$w_{7,0}$	25806
$w_{7,1}$	$2\,(6006\,c_1 - 6240\,c_2 - 11\,(-4095 + 2730\,c_3 + 3136\,c_4 - 234\,c_5) + 61152\,c_6)$
$w_{7,2}$	$-16692\,c_1 - 98\,(-1001 + 1326\,c_3 + 154\,c_5)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{7,3}$	$2\ (2574\, c_1 - 34496\, c_2 - 39\ (7\ (22\, c_3 + 224\, c_4 - 55\ (3 + 2\, c_5)) + 160\, c_6))$
$w_{7,4}$	$135212\, c_1 - 26\ (-3157 + 1038\, c_3 + 4074\, c_5)$
$w_{7,5}$	$2\ (6006\, c_1 + 6240\, c_2 - 30030\, c_3 + 11\ (4095 + 3136\, c_4 + 234\, c_5) - 61152\, c_6)$
$w_{7,6}$	$-15092\, c_1 + 26\ (3773 + 642\, c_3 + 4998\, c_5)$
$w_{7,7}$	$2\ (2574\, c_1 + 34496\, c_2 + 39\ (1155 - 154\, c_3 + 1568\, c_4 + 770\, c_5 + 160\, c_6))$
$w_{7,8}$	$26\ (3157 + 1038\, c_1 + 4074\, c_3) + 135212\, c_5$
$w_{7,9}$	$78\ (1155 + 770\, c_1 + 1568\, c_2 - 66\, c_3 - 160\, c_4 + 154\, c_5) + 68992\, c_6$
$w_{7,10}$	$129948\, c_1 + 7546\ (13 + 2\, c_3) - 16692\, c_5$
$w_{7,11}$	12870
$w_{7,12}$	$82082 - 105924\, c_1 - 135212\, c_3 + 26988\, c_5$
$w_{7,13}$	$78\ (1155 + 770\, c_1 - 1568\, c_2 - 66\, c_3 + 160\, c_4 + 154\, c_5) - 68992\, c_6$
$w_{7,14}$	-33
d_8	$\frac{1}{46126080}$
$w_{8,0}$	$462\ (257 + 256\, c_1 + c_5)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,1}$	$-14\big(572\,c_1{}^2 + 11\,(-4053 + 780\,c_3 + 2048\,c_4 + 3267\,c_5) + 4\,\big(-39\,c_2\,(-25 + 14\,c_3 + 11\,c_5) + 2\,\big(416\,c_3{}^2 + (143\,c_4 - 2400\,c_5)\,c_5 + 39\,c_3\,(22\,c_4 + 8\,c_5 - 15\,c_6)\big)\big) + 12\,c_1\,(-4785 + 286\,c_2 + 507\,c_3 + 2354\,c_4 + 3200\,c_5 - 3718\,c_6) - 39936\,c_6\big)$
$w_{8,2}$	$14\,\big(47509 - 44148\,c_3 - 26983\,c_5 + 4\,\big(8418\,c_1{}^2 + c_1\,(19455 - 11869\,c_3 - 9250\,c_5) - 832\,\big(c_3{}^2 + c_3\,c_5 - c_5{}^2\big)\big)\big)$
$w_{8,3}$	$-14\,\big(38400\,c_1{}^2 + 88\,c_2\,(256 + 78\,c_3 + 13\,c_5) + 4\,c_1\,(7062\,c_2 - 689\,c_3 + 11154\,c_4 - 6607\,c_5 + 858\,(-11 + c_6)) + 3\,(-14861 + 13312\,c_4 + 1991\,c_5 + 1300\,c_6) + 52\,\big(22\,c_3{}^2 + c_3\,(-33 + 90\,c_4 - 64\,c_5 - 42\,c_6) - 3\,c_5\,(30\,c_5 + 11\,c_6)\big)\big)$
$w_{8,4}$	$2\,\big(278795 + 287060\,c_1{}^2 - 343873\,c_5 + 4\,c_3\,(-50133 + 9516\,c_3 + 28501\,c_5) - 4\,c_1\,(-160391 + 59602\,c_3 + 57525\,c_5)\big)$
$w_{8,5}$	$-14\,\big(572\,c_1{}^2 + 11\,(-4053 + 780\,c_3 - 2048\,c_4 + 3267\,c_5) + 4\,\big(39\,c_2\,(-25 + 14\,c_3 + 11\,c_5) + 2\,\big(416\,c_3{}^2 - c_5\,(143\,c_4 + 2400\,c_5) - 39\,c_3\,(22\,c_4 - 8\,c_5 - 15\,c_6)\big)\big) - 12\,c_1\,(4785 + 286\,c_2 - 507\,c_3 + 2354\,c_4 - 3200\,c_5 - 3718\,c_6) + 39936\,c_6\big)$
$w_{8,6}$	$-14\,\big(-47509 + 20164\,c_1{}^2 + c_3\,(16836 - 20164\,c_5) + 4\,c_1\,(-11869 + 9250\,c_3 - 12701\,c_5) - 30673\,c_5\big)$
$w_{8,7}$	$-14\,\big(38400\,c_1{}^2 - 88\,c_2\,(256 + 78\,c_3 + 13\,c_5) - 3\,(14861 + 13312\,c_4 - 1991\,c_5 + 1300\,c_6) - 4\,c_1\,(7062\,c_2 + 689\,c_3 + 11154\,c_4 + 6607\,c_5 + 858\,(11 + c_6)) + 52\,\big(22\,c_3{}^2 - c_3\,(33 + 90\,c_4 + 64\,c_5 - 42\,c_6) + 3\,c_5\,(-30\,c_5 + 11\,c_6)\big)\big)$
$w_{8,8}$	$2\,\big(278795 + 230100\,c_3 + 135223\,c_5 + 4\,\big(81234\,c_1{}^2 + c_1\,(148275 + 60125\,c_3 + 31148\,c_5) + 52\,\big(233\,c_3{}^2 + 416\,c_3\,c_5 + 233\,c_5{}^2\big)\big)\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{8,9}$	$14\left(44583 + 7228\,c_1{}^2 + 21384\,c_3 - 2184\,c_3{}^2 + 312\,c_2\,(128 + 15\,c_3) - 3900\,c_4 + 2184\,c_3\,c_4 - 16269\,c_5 - 18056\,c_3\,c_5 + 1716\,c_4\,c_5 + 88\,(256 + 78\,c_3 + 13\,c_5)\,c_6 + 4\,c_1\,(16929 + 11154\,c_2 + 10432\,c_3 - 858\,c_4 + 1027\,c_5 + 7062\,c_6)\right)$
$w_{8,10}$	$14\left(47509 + 47476\,c_1{}^2 + 3328\,c_3{}^2 + c_3\,(13508 - 3328\,c_5) + (3361 - 16836\,c_5)\,c_5 + 4\,c_1\,(26283 + 5873\,c_3 + 8418\,c_5)\right)$
$w_{8,11}$	$14\,(59103 + 90880\,c_1 - 6656\,c_3 - 38433\,c_5)$
$w_{8,12}$	$-2\left(-278795 + 278564\,c_1{}^2 + 86528\,c_3{}^2 + 4\,c_3\,(62249 + 21632\,c_5) + c_5\,(-86759 + 162468\,c_5) + 4\,c_1\,(83881\,c_3 - 40617\,(1 + 2\,c_5))\right)$
$w_{8,13}$	$14\left(44583 + 7228\,c_1{}^2 + 21384\,c_3 + 3900\,c_4 - 16269\,c_5 - 22528\,c_6 - 4\,c_1\,(-16929 + 11154\,c_2 - 10432\,c_3 - 858\,c_4 - 1027\,c_5 + 7062\,c_6) - 4\left(546\,c_3{}^2 + 78\,c_2\,(128 + 15\,c_3) + 546\,c_3\,c_4 + 4514\,c_3\,c_5 + 429\,c_4\,c_5 + 286\,(6\,c_3 + c_5)\,c_6\right)\right)$
$w_{8,14}$	$231\,(1 + c_5)$
d_9	$\dfrac{1}{322882560}$
$w_{9,0}$	$66\,(12561 + 12480\,c_2 - 49\,c_3)$
$w_{9,1}$	$-2\left(88088\,c_1{}^2 + 168168\,c_2{}^2 + 7\,c_3\,(164109 + 35880\,c_3 - 149884\,c_5) + 11\,(-197253 + 56\,c_4\,(1792 + 143\,c_4) + 93816\,c_5) + 52\,c_1\,(1408\,c_2 + 7\,(231 - 608\,c_3 + 883\,c_5)) + 4\,c_2\,(81536\,c_3 + 3\,(-224832 + 103873\,c_4 + 156800\,c_5 - 195000\,c_6)) + 34944\,(-56 + 11\,c_4)\,c_6 - 251160\,c_6{}^2\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{9,2}$	$2\left(107016\,c_1{}^2 - 8\,c_2\,(296725\,c_3 + 49\,(-7623 + 2647\,c_5)) + 4\,c_1\,(195186 + 412482\,c_2 - 249214\,c_3 + 21021\,c_4 - 40768\,c_5 + 26754\,c_6) + 49\,(11\,(4235 + 52\,c_4 - 1228\,c_5) - 572\,c_5\,(2\,c_4 + c_5 + 12\,c_6) + c_3\,(4108\,c_5 - 3\,(19567 + 1560\,c_6)))\right)$
$w_{9,3}$	$-2\left(133224\,c_1{}^2 + 21\,(-103323 + 30679\,c_3 + 93184\,c_4 - 20020\,c_5) + 8\,c_1\,(128997 + 235200\,c_2 - 133651\,c_3 + 27664\,c_5) + 199680\,c_6 + 4\left(350658\,c_2{}^2 - 26\,c_2\,(17248 + 704\,c_3 - 22066\,c_4 + 3136\,c_5 - 5313\,c_6) - 91\left(1207\,c_3\,c_5 - 187\,c_5{}^2 - 261\,c_4\,c_6 + 366\,c_6{}^2\right)\right)\right)$
$w_{9,4}$	$-2\left(-2026409 + 163072\,c_1{}^2 + 2192721\,c_3 + 28028\,c_4 + 1610700\,c_5 - 28\,c_1\,(62249 + 68674\,c_2 + 4368\,c_3 + 2002\,c_4 + 1001\,c_5 + 12012\,c_6) + 4\left(2\,c_2\,(-343497 + 284319\,c_3 + 232661\,c_5) - 49\left(4800\,c_3{}^2 + 429\,c_3\,c_4 + 143\,c_5{}^2 + 78\,(7\,c_3 - 15\,c_5)\,c_6\right)\right)\right)$
$w_{9,5}$	$-2\left(88088\,c_1{}^2 - 168168\,c_2{}^2 - 77\,(28179 + 8\,c_4\,(1792 + 143\,c_4)) + 7\,c_3\,(164109 + 35880\,c_3 - 149884\,c_5) + 1031976\,c_5 + 52\,c_1\,(1408\,c_2 + 7\,(231 - 608\,c_3 + 883\,c_5)) - 34944\,(-56 + 11\,c_4)\,c_6 + 251160\,c_6{}^2 + 4\,c_2\,(-774336 + 81536\,c_3 - 311619\,c_4 + 470400\,c_5 + 585000\,c_6)\right)$
$w_{9,6}$	$-2\left(163072\,c_1{}^2 - 3\,c_3\,(-497947 + 313600\,c_3 - 28028\,c_4) + 8\,c_2\,(-373527 + 206241\,c_3 - 296725\,c_5) - 49\,(46585 + 572\,c_4 + 52\,c_5\,(849 + 11\,c_5)) + 15288\,(7\,c_3 - 15\,c_5)\,c_6 + 196\,c_1\,(3377 + 5294\,c_2 - 624\,c_3 + 286\,c_4 - 143\,c_5 + 1716\,c_6)\right)$
$w_{9,7}$	$2\left(-133224\,c_1{}^2 - 8\,c_1\,(128997 + 235200\,c_2 - 133651\,c_3 + 27664\,c_5) + 3\,(723261 - 214753\,c_3 + 652288\,c_4 + 140140\,c_5 + 66560\,c_6) + 4\left(350658\,c_2{}^2 + c_2\,(18304\,c_3 + 573716\,c_4 + 3136\,(319 + 26\,c_5) + 138138\,c_6) + 91\left(1207\,c_3\,c_5 - 187\,c_5{}^2 + 261\,c_4\,c_6 - 366\,c_6{}^2\right)\right)\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{9,8}$	$2\left(107016\,c_1{}^2+8\,c_2\,(343497+232661\,c_3+240359\,c_5)+4\,c_1\,(369906+568638\,c_2-249214\,c_3-21021\,c_4-40768\,c_5-26754\,c_6)+7\,\left(c_3\,(128229+28756\,c_5+32760\,c_6)+11\,\left(26317+22636\,c_5+364\,\left(-c_4+2\,c_4\,c_5-c_5{}^2+12\,c_5\,c_6\right)\right)\right)\right)$
$w_{9,9}$	$-2\left(221312\,c_1{}^2-2206776\,c_2{}^2+3\,c_3\,(-101211+327040\,c_3)-364\,c_1\,(896\,c_2+33\,(35+32\,c_3-13\,c_5))+77\,(-28179+52\,c_5\,(21+47\,c_5)-14336\,c_6)-8\,c_2\,(606816+235200\,c_3-21021\,c_4-9152\,c_5+206724\,c_6)-52\,\left(3003\,c_4{}^2+7392\,c_6{}^2+32\,c_4\,(-120+133\,c_6)\right)\right)$
$w_{9,10}$	$2\left(56056\,c_1{}^2-539\,(-4235+95\,c_3-52\,c_4)+780744\,c_5+196\,\left(-1170\,c_3{}^2+286\,c_3\,c_4-4657\,c_3\,c_5+429\,c_4\,c_5+2\,c_2\,(7623+2647\,c_3+4209\,c_5)+78\,(22\,c_3+7\,c_5)\,c_6\right)+52\,c_1\,(45650\,c_2+49\,(849+64\,c_3+53\,c_5+90\,c_6))\right)$
$w_{9,11}$	$2\,(2865423+442624\,c_1+4526336\,c_2-1960623\,c_3+256256\,c_5)$
$w_{9,12}$	$2\left(56056\,c_1{}^2-77\,(-26317+31897\,c_3+364\,c_4)+1479624\,c_5-52\,c_1\,(35794\,c_2+7\,(4425-448\,c_3-371\,c_5+630\,c_6))-28\,\left(c_2\,(-98142+68674\,c_3-81234\,c_5)+7\,\left(1170\,c_3{}^2+286\,c_3\,c_4+4657\,c_3\,c_5+429\,c_4\,c_5+78\,(22\,c_3+7\,c_5)\,c_6\right)\right)\right)$
$w_{9,13}$	$-2\left(221312\,c_1{}^2+2206776\,c_2{}^2+3\,c_3\,(-101211+327040\,c_3)-364\,c_1\,(896\,c_2+33\,(35+32\,c_3-13\,c_5))-8\,c_2\,(117600+235200\,c_3+21021\,c_4-9152\,c_5-206724\,c_6)+77\,(-28179+52\,c_5\,(21+47\,c_5)+14336\,c_6)+52\,\left(3003\,c_4{}^2+7392\,c_6{}^2+32\,c_4\,(-120+133\,c_6)\right)\right)$
$w_{9,14}$	$33\,(17+64\,c_2-49\,c_3)$
d_{10}	$\dfrac{1}{322882560}$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{10,0}$	$3234\,(257 + c_1 + 256\,c_3)$
$w_{10,1}$	$-2\,\big(156\,c_2\,(1335 + 1078\,c_3 - 686\,c_5) + c_1\,(84084\,c_2 + 347724\,c_3 - 49\,(-16269 + 1144\,c_4 + 18056\,c_5)) + 49\,\big(-44583 + 22528\,c_4 + 21384\,c_5 - 39936\,c_6 + 4\,\big(1521\,{c_3}^2 + 78\,c_5\,(22\,c_4 + 7\,c_5 - 15\,c_6) + c_3\,(7062\,c_4 + 8768\,c_5 - 33\,(383 + 338\,c_6))\big)\big)\big)$
$w_{10,2}$	$-14\,\big(-315403 + 162468\,{c_1}^2 + 351780\,{c_3}^2 + c_1\,(13081 - 235704\,c_3 - 86528\,c_5) + 4\,c_5\,(23639 + 21632\,c_5) + 4\,c_3\,(-29463 + 52265\,c_5)\big)$
$w_{10,3}$	$2\,\big(940800\,{c_1}^2 + 147\,(14861 + 20284\,c_3 - 13312\,c_4 + 2860\,c_5) - 208260\,c_6 + 49\,c_1\,(1144\,c_2 - 3\,(11979 + 12800\,c_3 - 832\,c_5 + 572\,c_6)) - 4\,\big(1078\,c_2\,(256 + 321\,c_3 + 78\,c_5) - 13\,\big(2277\,{c_3}^2 - 98\,c_5\,(45\,c_4 + 32\,c_5 - 21\,c_6) - 49\,c_3\,(858\,c_4 - 139\,c_5 + 66\,c_6)\big)\big)\big)$
$w_{10,4}$	$14\,\big(295955 + 23296\,{c_1}^2 - 324936\,{c_3}^2 - 52\,c_5\,(4425 + 448\,c_5) - 156\,c_3\,(-1121 + 1661\,c_5) + c_1\,(154671 + 385464\,c_3 + 23296\,c_5)\big)$
$w_{10,5}$	$2\,\big(c_1\,(84084\,c_2 - 347724\,c_3 - 49\,(16269 + 1144\,c_4 - 18056\,c_5)) + 156\,c_2\,(1335 + 1078\,c_3 - 686\,c_5) + 49\,\big(44583 + 22528\,c_4 - 21384\,c_5 - 39936\,c_6 - 4\,\big(1521\,{c_3}^2 + 78\,c_5\,(-22\,c_4 + 7\,c_5 + 15\,c_6) + c_3\,(-12639 - 7062\,c_4 + 8768\,c_5 + 11154\,c_6)\big)\big)\big)$
$w_{10,6}$	$14\,\big(315403 + 48464\,{c_1}^2 - 235704\,{c_3}^2 + c_1\,(-208329 + 1872\,c_3 - 86528\,c_5) + 52\,c_5\,(5943 + 932\,c_5) + 156\,c_3\,(2091 + 2011\,c_5)\big)$
$w_{10,7}$	$2\,\big(940800\,{c_1}^2 + 4312\,c_2\,(256 + 321\,c_3 + 78\,c_5) + 147\,(14861 + 13312\,c_4 + 2860\,c_5) - 49\,c_1\,(35937 + 1144\,c_2 + 38400\,c_3 - 2496\,c_5 - 1716\,c_6) + 208260\,c_6 + 4\,\big(29601\,{c_3}^2 - 1274\,c_5\,(-45\,c_4 + 32\,c_5 + 21\,c_6) + 49\,c_3\,(15213 + 11154\,c_4 + 1807\,c_5 + 858\,c_6)\big)\big)$
$w_{10,8}$	$-14\,\big(-295955 - 123367\,c_1 + 117852\,{c_1}^2 - 622668\,c_3 - 324936\,c_1\,c_3 - 259116\,{c_3}^2 - 4\,(62249 + 5824\,c_1 + 72727\,c_3)\,c_5 - 23296\,{c_5}^2\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{10,9}$	$2\left(2184567 + 229320\,c_1{}^2 + 3945480\,c_3 + 1881600\,c_3{}^2 - 208260\,c_4 - 168168\,c_3\,c_4 - 84084\,c_5 - 11388\,c_3\,c_5 + 107016\,c_4\,c_5 - 56056\,c_5{}^2 + 15288\,c_2\,(128 + 143\,c_3 + 15\,c_5) + 4312\,(256 + 321\,c_3 + 78\,c_5)\,c_6 - 49\,c_1\,(5973 + 13116\,c_3 + 1716\,c_4 + 3328\,c_5 + 1144\,c_6)\right)$
$w_{10,10}$	$14\left(315403 + 160596\,c_3{}^2 + c_1\,(195263 + 303316\,c_3 - 114004\,c_5) + 12\,c_5\,(8391 + 3172\,c_5) + 4\,c_3\,(130361 + 80558\,c_5)\right)$
$w_{10,11}$	$5792094 - 3766434\,c_1 + 9199104\,c_3 + 652288\,c_5$
$w_{10,12}$	$-14\,(-295955 + 4\,c_3\,(-35893 + 66903\,c_3 - 75410\,c_5) - 217692\,c_5 + c_1\,(324425 + 282412\,c_3 + 141148\,c_5))$
$w_{10,13}$	$2\left(229320\,c_1{}^2 + 3945480\,c_3 + 208260\,c_4 - 49\,c_1\,(5973 + 13116\,c_3 - 1716\,c_4 + 3328\,c_5 - 1144\,c_6) - 539\,(-4053 + 156\,c_5 + 2048\,c_6) - 4\left(-470400\,c_3{}^2 + 3822\,c_2\,(128 + 143\,c_3 + 15\,c_5) + c_3\,(-42042\,c_4 + 2847\,c_5 + 346038\,c_6) + 1274\,c_5\,(21\,c_4 + 11\,(c_5 + 6\,c_6))\right)\right)$
$w_{10,14}$	$1617\,(1 + c_1)$
d_{11}	$\dfrac{1}{322882560}$
$w_{11,0}$	$66\,(12561 + 49\,c_1 + 12480\,c_4)$
$w_{11,1}$	$2\left(133224\,c_2{}^2 + c_1\,(644259 + 439348\,c_3 - 73216\,c_4 - 1069208\,c_5) - 11\,(7\,(-28179 + 52\,c_3\,(105 + 17\,c_3) - 23296\,c_4) + 93816\,c_5) - 56\left(25047\,c_4{}^2 + 208\,c_3\,(28\,c_4 - 19\,c_5) + 33600\,c_4\,c_5 + 2379\,c_5{}^2\right) - 156\,c_2\,(1280 + 3542\,c_4 - 609\,c_6) + 1956864\,c_6 + 2294864\,c_4\,c_6\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{11,2}$	$2\left(940800\,c_1^2 + 28028\,c_3^2 - 52\,c_3\,(41601 + 4410\,c_2 + 45650\,c_4 + 539\,c_5) + 3\,c_1\,(497947 + 35672\,c_2 + 549976\,c_4 - 40768\,c_5 - 28028\,c_6) - 49\,(8\,c_4\,(-7623 + 2647\,c_5) + 4\,c_5\,(3377 + 1716\,c_2 + 832\,c_5 - 286\,c_6) + 11\,(-4235 + 52\,c_6))\right)$
$w_{11,3}$	$-2\left(981120\,c_1^2 + 384384\,c_2^2 - 21\,(103323 + 44800\,c_4 + 20020\,c_5) + 3\,c_1\,(101211 + 627200\,c_4 + 128128\,c_5) + 224\,c_2\,(4928 + 7383\,c_4 - 988\,c_6) + 199680\,c_6 + 52\left(3619\,c_3^2 + 42438\,c_4^2 - 6272\,c_4\,c_5 + 4256\,c_5^2 - 11\,c_3\,(147 + 128\,c_4 + 273\,c_5) + 3234\,c_4\,c_6 + 3003\,c_6^2\right)\right)$
$w_{11,4}$	$-2\left(229320\,c_1^2 + 88\left(-31227\,c_4 + 21151\,c_4\,c_5 - 637\,c_5^2\right) - 7\,c_1\,(350867 + 48048\,c_2 - 130396\,c_3 + 274696\,c_4 - 23296\,c_5 - 8008\,c_6) + 4\,c_3\,(369906 - 26754\,c_2 + 568638\,c_4 + 33761\,c_5 + 21021\,c_6) + 7\,(780\,(295 + 42\,c_2)\,c_5 - 11\,(26317 + 364\,c_6))\right)$
$w_{11,5}$	$2\left(-133224\,c_2^2 + c_1\,(644259 + 439348\,c_3 - 73216\,c_4 - 1069208\,c_5) - 11\,(7\,(-28179 + 52\,c_3\,(105 + 17\,c_3) - 51968\,c_4) + 93816\,c_5) - 56\left(-25047\,c_4^2 + 208\,c_3\,(28\,c_4 - 19\,c_5) + 33600\,c_4\,c_5 + 2379\,c_5^2\right) + 156\,c_2\,(1280 + 3542\,c_4 - 609\,c_6) - 1956864\,c_6 - 2294864\,c_4\,c_6\right)$
$w_{11,6}$	$-2\left(229320\,c_1^2 + 780744\,c_3 + 4\,(3822\,c_2\,(7\,c_3 - 15\,c_5) - 286\,c_5\,(2075\,c_4 + 49\,c_5) + 49\,c_3\,(8418\,c_4 + 689\,c_5 - 429\,c_6)) + 49\,c_1\,(-1045 + 6864\,c_2 + 18628\,c_3 + 21176\,c_4 + 3328\,c_5 - 1144\,c_6) - 49\,(46585 + 60984\,c_4 + 44148\,c_5 - 572\,c_6)\right)$
$w_{11,7}$	$2\left(-981120\,c_1^2 - 4004\,c_3\,(-21 + 47\,c_3) + 21\,(103323 + 231168\,c_4 + 20020\,c_5) - 3\,c_1\,(101211 + 627200\,c_4 + 128128\,c_5) + 199680\,c_6 + 4\left(96096\,c_2^2 + 56\,c_2\,(4928 + 7383\,c_4 - 988\,c_6) + 13\left(1408\,c_3\,c_4 + 42438\,c_4^2 + 3003\,c_3\,c_5 + 6272\,c_4\,c_5 - 4256\,c_5^2 + 3234\,c_4\,c_6 + 3003\,c_6^2\right)\right)\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{11,8}$	$2\left(940800\,c_1^2+4\left(7007\,c_3^2+13\,c_3\,(30975+4410\,c_2+35794\,c_4-539\,c_5)+14\,c_5\,(6006\,c_2+34337\,c_4-2912\,c_5-1001\,c_6)\right)-3\,c_1\,(-730907+35672\,c_2-758184\,c_4+40768\,c_5-28028\,c_6)+77\,(26317+35688\,c_4+22636\,c_5+364\,c_6)\right)$
$w_{11,9}$	$2\left(-251160\,c_1^2+2184\,c_2\,(896+115\,c_2)+1031976\,c_3+2697984\,c_4+7\,c_1\,(164109+149884\,c_3+46592\,c_4-31616\,c_5)-77\,(-28179+1092\,c_5-14336\,c_6)+4\left(7\,c_3\,(67200\,c_4+11479\,c_5)+312\,c_2\,(1875\,c_4+308\,c_6)-11\left(3822\,c_4^2+c_4\,(1664\,c_5-28329\,c_6)+2002\left(c_5^2+c_6^2\right)\right)\right)\right)$
$w_{11,10}$	$2\left(2282665+2988216\,c_4+780744\,c_5+c_1\,(2876349+229320\,c_2+201292\,c_3+2373800\,c_4+996856\,c_5)+196\left(-143\,c_3^2+c_3\,(3377+1716\,c_2+5294\,c_4+832\,c_5-286\,c_6)+3\,c_5\,(182\,c_2+2806\,c_4+182\,c_5-143\,c_6)\right)-28028\,c_6\right)$
$w_{11,11}$	$2\,(2865423+1960623\,c_1-256256\,c_3+4526336\,c_4+442624\,c_5)$
$w_{11,12}$	$-2\left(-2026409-2747976\,c_4+c_1\,(897603+229320\,c_2-201292\,c_3+1861288\,c_4-996856\,c_5)-1479624\,c_5-28028\,c_6+28\left(1001\,c_3^2+c_3\,(62249+12012\,c_2+68674\,c_4-5824\,c_5-2002\,c_6)-3\,c_5\,(-1274\,c_2+27078\,c_4+1274\,c_5+1001\,c_6)\right)\right)$
$w_{11,13}$	$-2\left(251160\,c_1^2+251160\,c_2^2-3097344\,c_4-7\,c_1\,(164109+149884\,c_3+46592\,c_4-31616\,c_5)+1248\,c_2\,(1568+1875\,c_4+308\,c_6)+77\,(-28179+1092\,c_5+14336\,c_6)-4\left(c_3\,(257994+470400\,c_4+80353\,c_5)+11\left(3822\,c_4^2-2002\,(c_5-c_6)\,(c_5+c_6)-c_4\,(1664\,c_5+28329\,c_6)\right)\right)\right)$
$w_{11,14}$	$33\,(17+49\,c_1+64\,c_4)$
d_{12}	$\dfrac{1}{46126080}$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{12,0}$	$-462\,(-257 + c_3 - 256\,c_5)$
$w_{12,1}$	$-14\,\big(1144\,{c_1}^2 - 11\,(4053 + 543\,c_3 - 2048\,c_4 + 3432\,c_5) + 4\,\big(-1170\,{c_3}^2 + 39\,c_2\,(25 + 11\,c_3 + 22\,c_5) + c_3\,(-286\,c_4 + 6607\,c_5) + 6\,c_5\,(1177\,c_4 + 1600\,c_5 - 1859\,c_6)\big) - 39936\,c_6 + 52\,c_1\,(33 + 42\,c_2 - 64\,c_3 - 132\,c_4 + 53\,c_5 + 90\,c_6)\big)$
$w_{12,2}$	$14\,\big(47509 + 16836\,c_1 - 30673\,c_3 + 20164\,c_1\,c_3 + 4\,(11869 + 9250\,c_1 - 12701\,c_3)\,c_5 - 20164\,{c_5}^2\big)$
$w_{12,3}$	$-14\,\big(22528\,c_2 - 3\,(14861 + 5423\,c_3 - 13312\,c_4 + 22572\,c_5 - 1300\,c_6) + 4\,\big(546\,{c_1}^2 - 22\,c_2\,(13\,c_3 - 321\,c_5) + 13\,(79\,c_3 + 858\,c_4 - 139\,c_5)\,c_5 + 429\,(c_3 + 2\,c_5)\,c_6 + 2\,c_1\,(2673 - 858\,c_2 + 2257\,c_3 - 585\,c_4 + 5216\,c_5 + 273\,c_6)\big)\big)$
$w_{12,4}$	$-2\,\big(-278795 + 86528\,{c_1}^2 + 86759\,c_3 + 162468\,{c_3}^2 + 4\,c_1\,(-62249 + 21632\,c_3 - 83881\,c_5) + 162468\,(-1 + 2\,c_3)\,c_5 + 278564\,{c_5}^2\big)$
$w_{12,5}$	$-14\,\big(1144\,{c_1}^2 - 11\,(4053 + 543\,c_3 + 2048\,c_4 + 3432\,c_5) + 39936\,c_6 - 52\,c_1\,(-33 + 42\,c_2 + 64\,c_3 - 132\,c_4 - 53\,c_5 + 90\,c_6) - 4\,\big(1170\,{c_3}^2 + 39\,c_2\,(25 + 11\,c_3 + 22\,c_5) - c_3\,(286\,c_4 + 6607\,c_5) - 6\,c_5\,(-1177\,c_4 + 1600\,c_5 + 1859\,c_6)\big)\big)$
$w_{12,6}$	$14\,\big(47509 + 3328\,{c_1}^2 - 16836\,{c_3}^2 - 4\,c_1\,(3377 + 832\,c_3 + 5873\,c_5) + 4\,c_5\,(26283 + 11869\,c_5) - c_3\,(3361 + 33672\,c_5)\big)$
$w_{12,7}$	$-14\,\big(2184\,{c_1}^2 + 8\,c_1\,(2673 + 858\,c_2 + 2257\,c_3 + 585\,c_4 + 5216\,c_5 - 273\,c_6) - 3\,(14861 + 5423\,c_3 + 13312\,c_4 + 22572\,c_5 + 1300\,c_6) + 4\,(22\,c_2\,(-256 + 13\,c_3 - 321\,c_5) + 13\,(79\,c_3 - 858\,c_4 - 139\,c_5)\,c_5 - 429\,(c_3 + 2\,c_5)\,c_6)\big)$
$w_{12,8}$	$2\,\big(278795 + 38064\,{c_1}^2 + 4\,c_5\,(160391 + 71765\,c_5) + c_3\,(343873 + 230100\,c_5) + 4\,c_1\,(28501\,c_3 + 17\,(2949 + 3506\,c_5))\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{12,9}$	$-14\left(-3\,(14861+13312\,c_2+11979\,c_3-1300\,c_4+19140\,c_5)-22528\,c_6 + 4\left(832\,{c_1}^2-4800\,{c_3}^2+143\,c_5\,(-78\,c_2+6\,c_4+c_5)-7062\,c_5\,c_6 + 39\,c_1\,(-55+30\,c_2+16\,c_3+14\,c_4-39\,c_5+44\,c_6)+c_3\,(429\,c_4-9600\,c_5+286\,c_6)\right)\right)$
$w_{12,10}$	$-14\left(-47509-26983\,c_3-77820\,c_5 + 4\left(832\,{c_1}^2+13\,c_1\,(-849+64\,c_3-913\,c_5)-2\,(c_3+c_5)\,(416\,c_3+4209\,c_5)\right)\right)$
$w_{12,11}$	$14\,(59103+6656\,c_1+38433\,c_3+90880\,c_5)$
$w_{12,12}$	$2\left(278795+48464\,{c_1}^2+48464\,{c_3}^2+36\,c_5\,(16475+9026\,c_5) - c_3\,(135223+124592\,c_5)+52\,c_1\,(1664\,c_3-25\,(177+185\,c_5))\right)$
$w_{12,13}$	$-14\left(39936\,c_2-3\,(14861+11979\,c_3+1300\,c_4+19140\,c_5)+22528\,c_6 + 4\left(832\,{c_1}^2-4800\,{c_3}^2+143\,c_5\,(78\,c_2-6\,c_4+c_5)+7062\,c_5\,c_6 - 39\,c_1\,(55+30\,c_2-16\,c_3+14\,c_4+39\,c_5+44\,c_6)-c_3\,(429\,c_4+9600\,c_5+286\,c_6)\right)\right)$
$w_{12,14}$	$-231\,(-1+c_3)$
d_{13}	$\dfrac{1}{322882560}$
$w_{13,0}$	$66\,(12561+49\,c_5+12480\,c_6)$
$w_{13,1}$	$-2\left(-156156\,{c_2}^2+1664\,c_2\,(120+133\,c_4) + 11\,(7\,(-28179+5460\,c_3+128\,(112-39\,c_4)\,c_4)+27603\,c_5) - 4854528\,c_6+4\left(47047\,{c_1}^2+224\left(247\,{c_3}^2-429\,c_3\,c_5+1095\,{c_5}^2\right) - 143\,c_1\,(-147+273\,c_3-128\,c_6) + 14\,(3003\,c_2+5824\,c_3+29532\,c_4+33600\,c_5)\,c_6-551694\,{c_6}^2\right)\right)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{13,2}$	$2\,\big(-28028\,c_2\,(-1+2\,c_5)+49\,\big(55\,(847+19\,c_5) + 52\,\big(c_3\,(-849+22\,c_3+90\,c_4)+4\,(16\,c_3+33\,c_4)\,c_5-90\,{c_5}^2\big)\big) - 8\,(-373527+296725\,c_3+129703\,c_5)\,c_6 + 4\,c_1\,(195186+21021\,c_2-33761\,c_3-26754\,c_4+228193\,c_5+412482\,c_6)\big)$
$w_{13,3}$	$2\,\big(-88088\,{c_2}^2+21\,(103323+104\,c_4\,(-896+115\,c_4)+54703\,c_5) + 4\,c_1\,(-257994+80353\,c_3-262297\,c_5-470400\,c_6) + 308\,c_2\,(-3584+1248\,c_4-4047\,c_6)+2697984\,c_6 - 52\,\big(7\,\big(11\,c_3\,(-21+22\,c_3)-608\,c_3\,c_5+690\,{c_5}^2\big) - 8\,(176\,c_3-5625\,c_4+784\,c_5)\,c_6+3234\,{c_6}^2\big)\big)$
$w_{13,4}$	$-2\,\big(-2026409+28028\,{c_1}^2+1479624\,c_3-2548\,(11\,c_2\,(-1+3\,c_3)+42\,c_3\,(c_3-c_4)) +897603\,c_5+392\,(2543\,c_3-585\,c_4)\,c_5-2747976\,c_6+8\,(284319\,c_3+232661\,c_5)\,c_6 - 28\,c_1\,(62249+2002\,c_2+5824\,c_3-12012\,c_4-7189\,c_5+68674\,c_6)\big)$
$w_{13,5}$	$-2\,\big(188188\,{c_1}^2+156156\,{c_2}^2+11\,(-197253+896\,c_4\,(-112+39\,c_4)+27603\,c_5) -572\,c_1\,(-147+273\,c_3-128\,c_6)-940800\,c_6-104\,c_2\,(1920+2128\,c_4+1617\,c_6) + 4\,\big(55328\,{c_3}^2+245280\,{c_5}^2-3003\,c_3\,(-35+32\,c_5) + 56\,(1456\,c_3-7383\,c_4+8400\,c_5)\,c_6+551694\,{c_6}^2\big)\big)$
$w_{13,6}$	$-2\,\big(-28028\,c_2+780744\,c_3+196\,\big(143\,{c_1}^2+429\,c_2\,c_3-546\,c_3\,(c_3+c_4) + 2\,(2543\,c_3+585\,c_4)\,c_5+c_1\,(3377+286\,c_2-832\,c_3-1716\,c_4+1027\,c_5)\big) + 8\,(129703\,c_1+206241\,c_3-296725\,c_5)\,c_6-49\,(58701\,c_5+847\,(55+72\,c_6))\big)$
$w_{13,7}$	$2\,\big(88088\,{c_2}^2+4\,c_1\,(-257994+80353\,c_3-262297\,c_5-470400\,c_6) - 308\,c_2\,(-3584+1248\,c_4-4047\,c_6) + 3\,(723261+728\,(896-115\,c_4)\,c_4+382921\,c_5+1032448\,c_6) + 52\,\big(-7\,\big(11\,c_3\,(-21+22\,c_3)-608\,c_3\,c_5+690\,{c_5}^2\big) + 8\,(176\,c_3+5625\,c_4+784\,c_5)\,c_6+3234\,{c_6}^2\big)\big)$

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{13,8}$	$2\left(7\left(-4004\,c_2+230100\,c_3+11\left(26317+31897\,c_5+35688\,c_6\right)\right)+4\left(1274\left(11\,{c_3}^2+\left(11\left(c_2-6\,c_4\right)-45\,c_5\right)c_5+c_3\left(-45\,c_4+32\,c_5\right)\right)+2\left(232661\,c_3+240359\,c_5\right)c_6+c_1\left(369906-21021\,c_2-33761\,c_3+26754\,c_4+228193\,c_5+568638\,c_6\right)\right)\right)$
$w_{13,9}$	$-2\left(68068\,{c_1}^2+312\,c_4\left(640+427\,c_4\right)-364\,c_1\left(1155+608\,c_3-1207\,c_5+896\,c_6\right)-77\left(28179+8367\,c_5+51968\,c_6\right)+52\,c_2\left(1827\,c_4-4\left(9408+11033\,c_6\right)\right)+8\left(16653\,{c_3}^2+11\left(6279\,c_4+832\,c_5-15939\,c_6\right)c_6-c_3\left(128997+133651\,c_5+235200\,c_6\right)\right)\right)$
$w_{13,10}$	$2\left(28028\,{c_1}^2+1493841\,c_5+28028\,c_2\left(1+2\,c_3+3\,c_5\right)+41503\left(55+72\,c_6\right)+52\,c_1\left(-49\left(-849+11\,c_3+90\,c_4\right)+45650\,c_6\right)-196\left(832\,{c_3}^2+c_3\left(-3377+1716\,c_4-624\,c_5-5294\,c_6\right)-6\,c_5\left(-91\,c_4+800\,c_5+1403\,c_6\right)\right)\right)$
$w_{13,11}$	$2\left(2865423+256256\,c_1-442624\,c_3+1960623\,c_5+4526336\,c_6\right)$
$w_{13,12}$	$2\left(2026409+28028\,{c_1}^2+2192721\,c_5-28028\,c_2\left(1+2\,c_3+3\,c_5\right)+2747976\,c_6-52\,c_1\left(30975+539\,c_3-4410\,c_4+35794\,c_6\right)-28\left(5824\,{c_3}^2-6\,c_5\left(637\,c_4+5600\,c_5+13539\,c_6\right)+c_3\left(62249-12012\,c_4-4368\,c_5+68674\,c_6\right)\right)\right)$
$w_{13,13}$	$2\left(-68068\,{c_1}^2+1031976\,c_3+1092\,c_2\left(-1792+87\,c_4\right)+312\,c_4\left(640+427\,c_4\right)+364\,c_1\left(1155+608\,c_3-1207\,c_5+896\,c_6\right)+77\left(28179+8367\,c_5+23296\,c_6\right)+8\left(-16653\,{c_3}^2-11\,c_6\left(26078\,c_2-6279\,c_4+832\,c_5+15939\,c_6\right)+7\,c_3\left(19093\,c_5+33600\,c_6\right)\right)\right)$
$w_{13,14}$	$33\left(17+49\,c_5+64\,c_6\right)$
d_{14}	$\dfrac{1}{315315}$
$w_{14,0}$	1617

(*continues*)

Table C.7: Integral approximation I_j, $n = 14$. (*continued*)

d_j, w_{jk}	Expression
$w_{14,1}$	$-286\,c_1 - 780\,c_2 - 49\,(-231 + 26\,c_3 + 88\,c_4 + 150\,c_5 - 156\,c_6)$
$w_{14,2}$	$28\,(418 + 225\,c_1 - 299\,c_3 - 106\,c_5)$
$w_{14,3}$	$11319 - 7350\,c_1 - 4312\,c_2 + 286\,c_3 - 7644\,c_4 + 1274\,c_5 - 780\,c_6$
$w_{14,4}$	$7\,(1529 + 918\,c_1 - 1290\,c_3 - 910\,c_5)$
$w_{14,5}$	$-286\,c_1 + 780\,c_2 - 49\,(-231 + 26\,c_3 - 88\,c_4 + 150\,c_5 + 156\,c_6)$
$w_{14,6}$	$-28\,(-418 + 106\,c_1 + 225\,c_3 - 299\,c_5)$
$w_{14,7}$	$-7350\,c_1 + 4312\,c_2 + 286\,c_3 + 49\,(231 + 156\,c_4 + 26\,c_5) + 780\,c_6$
$w_{14,8}$	$7\,(1529 + 1290\,c_1 + 910\,c_3 + 918\,c_5)$
$w_{14,9}$	$11319 + 1274\,c_1 + 7644\,c_2 + 7350\,c_3 - 780\,c_4 - 286\,c_5 + 4312\,c_6$
$w_{14,10}$	$28\,(418 + 299\,c_1 + 106\,c_3 + 225\,c_5)$
$w_{14,11}$	17681
$w_{14,12}$	$-7\,(-1529 + 910\,c_1 + 918\,c_3 - 1290\,c_5)$
$w_{14,13}$	$11319 + 1274\,c_1 - 7644\,c_2 + 7350\,c_3 + 780\,c_4 - 286\,c_5 - 4312\,c_6$
$w_{14,14}$	0